Latent Dirichlet Allocation models considering emojis

Taikgun Song

0.0.1 abstract

XXX write later XXX

Contents

0.0.1 abstract 1

1 Introduction **1**

2 LDA **2**
2.1 LDA Equation goes here 2

3 Application **2**
3.1 Data Set and exploratory data analysis 2
3.2 Application 3
3.3 LDA on a raw data set 3
3.4 LDA without Unicode 4
3.5 LDA with name translated 5

4 Conclusion **6**

5 Appendix **7**

6 `emoji` package in R **7**
6.1 Emoji Data Set 7
6.2 Generate different type of encodings using Python 7
6.3 Scoring of Sentiment 7
6.4 Description of the Emoji Data set 7

1 Introduction

The development of Social Network Service(SNS) and User Generated Content(UGC) platforms have been providing a new channels for its users to communicate and share opinions with the fellow community members. Hundreds and thousands of new messages are shared every day on Twitter and Facebook; product reviews are posted on Amazon and Trip Advisor. The text contents of SNS and UGC is a raw information of Individual's perception and emotion. As one would expect, this distinct characteristic of SNS and UGC text data could be a valuable information in different academic disciplines and industries. As a result, different machine learning and natural language processing techniques were developed to evaluate text data.

Most natural language processing(NLP) techniques require preparation procedures to improve its performance. Deleting stop words(determiners, conjunctions, prepositions, pronouns, auxiliary verbs, modals, and quantifiers) is an example of this preceding step. The result of this step increases the accuracy of the NLP by reducing the noise generated from words that does not convey any contextual meaning. In this context, Emoji(a pictographic information that carries class of feelings) in the text data has been considered as a noise and was deleted prior to applying NLP techniques.

Although deleting Emoji Unicode before NLP is a standard operation, unlike deleting stop words, this does take away information that might have modest contribution to the context. Emoji, originally driven from Japanese word e(picture) + moji(character), is a pictograph that has become widely used on internet web pages and on SNS plaforms. Emoji, much like its close relative emoticon, could provide visual representation of not only solid objects, but also emotions through facial expressions and symbols related to feelings and moods. Communication via traditional text characters, such as words in alphabets, may not be as effective and efficient as Emojis when conveying emotions. For this reason, Emoji gained popularity after 1990 especially after cell phones and internet came to wide use. Therefore, simply deleting Emoji would reduce the information contained in the original text data.

In contrast to filtering Emoji characters out, reflecting Emoji information during the evaluation have benefits. First, eacho Emoji has pre-determined topic dimension set by the official organization. Therefore, crude topic matching may be accomplished using the Emoji information. Second, Emoji may be helpful for sentiment analysis. Being closely related to emotions and mood, sentiment analysis on Emoji will provide auxiliary information.

The `emoji` package in `R` was written to help the above analysis.

2 LDA

To address the importance or reflecting the emoji information in text data, Latent Dirichlet Allocation (LDA): a popular topic modeling method, was performed on twitter messages scraped online. LDA is a topic modeling method that allows words observed in documents to be explained by unobserved topics and that each word's creation is attributable to one of the document's topics.

LDA is based on the two following principles:

1. Every document is a mixture of topics
2. Every topic is a mixture of words

To illustrate, a news paper document may contain several topics such as "politics", "economy", "spots", "entertainment", and etc. For a given topic "politics", common words may be "government", "trump", "president", "congress", and etc.

LDA assumes that the probability of documents are rnadom mixture over unseen topics, and document i having topic k follows a dirichlet distribution with some parameter α. That is, if the probability of document i having topic k is denoted as $\theta_{i,k}$, then $\theta_i \sim Dir(\alpha)$. The second assumption says each topic is a mixture of words, and that the distribution of n^{th} word will follow a multinomial distribution conditioned on the topic z. The probability of word given a topic is denoted as β. Then β has a Dirichlet distribution with parameter η.

1. $\theta_i \sim Dir(\alpha), i = 1, \ldots, M$
2. $\theta_{i,k}$ is the probability that document $i \in \{1, \ldots, M\}$ has topic $k \in \{1, \ldots, K\}$.
3. z is word's topic drawn from a Multinomial distribution with parameter θ, i.e. $z \sim Multi(\theta)$
4. $\beta_k \sim Dir(\eta), k = 1, \ldots, K$
5. $\beta_{k,v}$ is the probability of word $v \in \{1, \ldots, V\}$ in topic $k \in \{1, \ldots, K\}$
6. w is a word drawn from a Multinomial distribution with parameter Z and β, i.e., $w \sim Multi(z, \beta)$.

The marginal distribution of word w given hyper parameter α and β is obtained by the following equation:

$$p(w|\alpha, \beta) = \int p(\theta|\alpha)(\prod_{v=1}^{V} \sum_{z_v} p(z_v|\theta)p(w_v|z_v, \beta))d\theta$$

where

Graphical display of LDA is given in Figure 1.

2.1 LDA Equation goes here

3 Application

3.1 Data Set and exploratory data analysis

Two samples of twitter messages with the following hastag #inlove and #hateher were scraped. The data set contains 944 #inlove messages, 1145 #hateher messages, and 1195 #marchscience messages. The proportion of Twitter messages containing Emoji characters per hashtag is illustraited in Table 1. 52.7% of the #inlove message strings, 29.3% of the #hateher message strings, and 7.8% of #marchscience message strings have one or more emoji information.

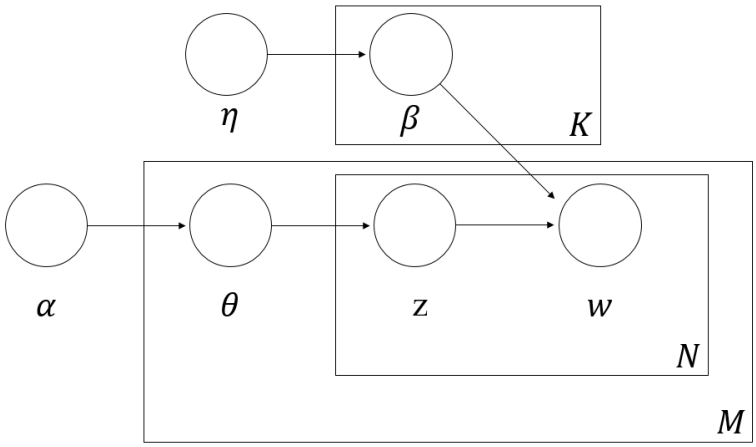


Figure 1: Graphical Model representation of LDA

Table 1: Proportion of Twitter messages with Emoji

	#inlove	#hateher	#marchscience
Proportion	0.5275	0.2926	0.07782

For hashtag #inlove, total number of 1188 Emojis were used from 182 unique emojis. For hashtag #hateher, 695 Emojis from 112 unique Emojis were used. For hashtag #sciencemarch, 202 Emojis from 102 unique Emojis were used (Note that there may be multiple Emojis per Twitter message). Top 5 frequently used Emojis per hashtag is given in Table 2.

#inlove	Emoji	Count	#hateher	Emoji	Count	#marchscience	Emoji	Count
U+1F60D	😍	297	U+1F602	😂	154	U+1F52C	🔬	13
U+2764	❤️	164	U+1F644	🙄	88	U+1F30E	🌎	11
U+1F495	💕	47	U+1F621	😡	40	U+1F44D	👍	9
U+1F618	😘	40	U+1F612	😒	38	U+1F680	🚀	8
U+2728	✨	26	U+1F62D	😭	36	U+1F30D	🌍	7

Table 2: Five most popular Emoji for each hastags

It is interesting to see “Face with tear of joy” as the most popular Emoji for hashtag #hateher. Although the name itself contains the word “joy”, some users of this Emoji adopted this pictogram to express their mixed feeling of love and hate at the same time.

3.2 Application

LDA was performed on the following three difference cases:

1. LDA on a raw data set
2. LDA on a data set with Unicode removed
3. LDA on a data set with Emoji translated to text

3.3 LDA on a raw data set

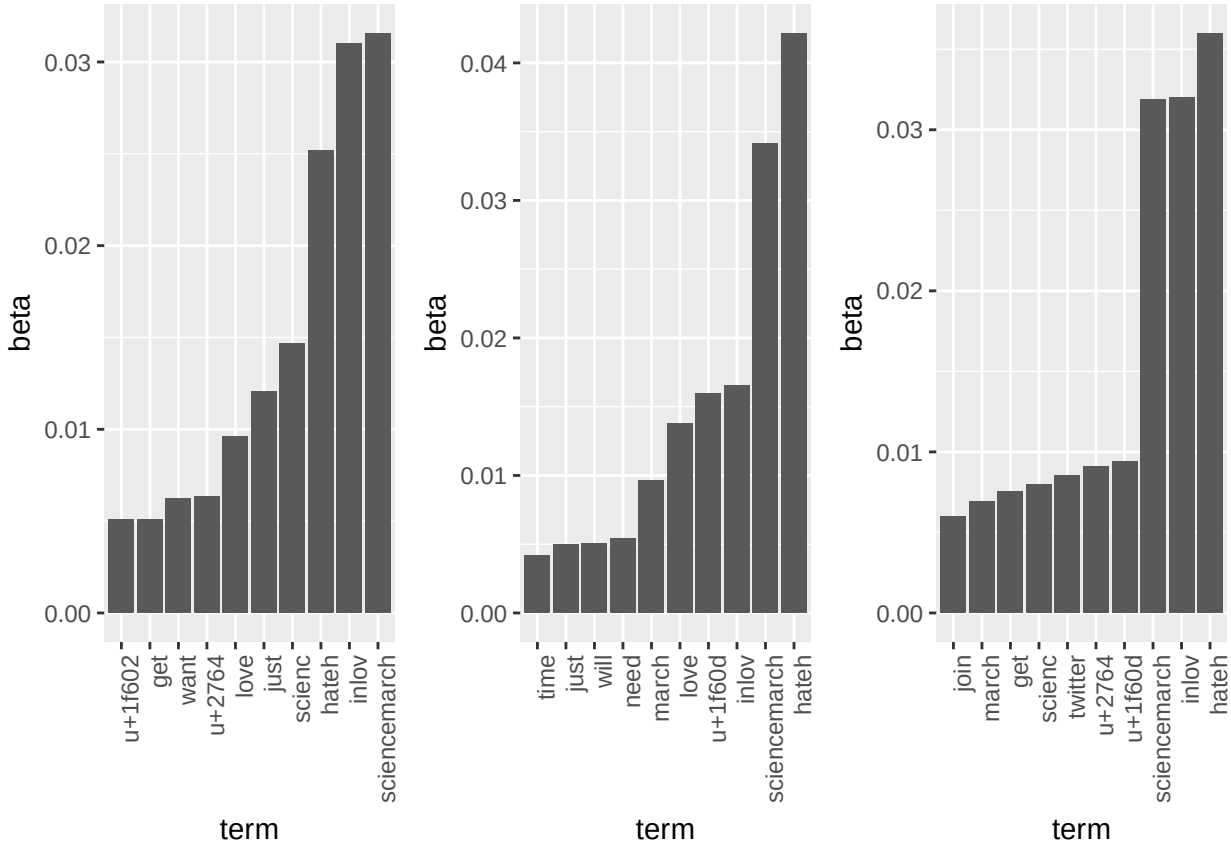
The second case was to run LDA on a raw data set. Stemming and stop word deletion were performed. Different number of topic dimensions were tested and the result of 4 topic dimension with 10 terms are provided in Table 3. Describe the output.

Table 3: Output LDA with the raw data

Topic 1	Topic 2	Topic 3
sciencemarch	hateh	hateh

Table 4: Word prob. given topic

1.term	1.beta	2.term	2.beta	3.term	3.beta
sciencemarch	0.03155	hateh	0.04214	hateh	0.03598
inlov	0.03098	sciencemarch	0.03417	inlov	0.03203
hateh	0.02517	inlov	0.01651	sciencemarch	0.03187
scienc	0.01465	u+1f60d	0.01597	u+1f60d	0.009386
just	0.01208	love	0.01377	u+2764	0.009117
love	0.009608	march	0.009667	twitter	0.008563
u+2764	0.006328	need	0.005389	scienc	0.007948
want	0.006217	will	0.005078	get	0.007516
get	0.005093	just	0.004977	march	0.006911
u+1f602	0.005081	time	0.004192	join	0.005993



3.4 LDA without Unicode

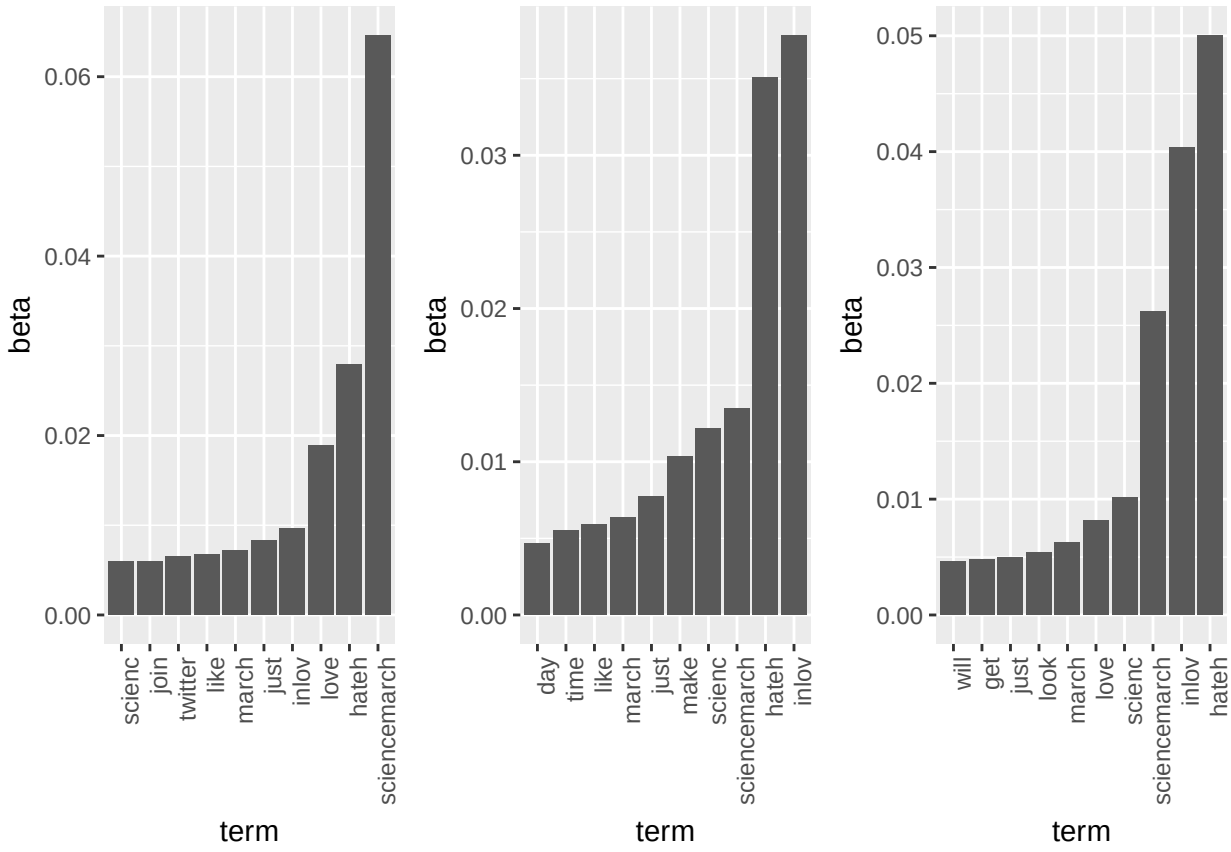
In most text mining examples, LDA is performed after removing the unicode information. For the first case, therefore, unicode characters were removed from the raw text data set. Then, the standard procedure of stemming and stop word deletion was performed to enhance the accuracy of LDA. `tm` package was used to conduct the above procedure.

Table 5: Output of LDA with the raw data without the Unicode

Topic 1	Topic 2	Topic 3
sciencemarch	inlov	hateh
hateh	hateh	inlov
love	sciencemarch	sciencemarch
inlov	scienc	scienc
just	make	love

Table 6: Word prob. given topic

1.term	1.beta	2.term	2.beta	3.term	3.beta
sciencemarch	0.0646	inlov	0.03782	hateh	0.05003
hateh	0.02794	hateh	0.03507	inlov	0.04038
love	0.01887	sciencemarch	0.01347	sciencemarch	0.02619
inlov	0.009616	scienc	0.01216	scienc	0.01016
just	0.008348	make	0.01033	love	0.008194
march	0.007189	just	0.007723	march	0.006298
like	0.006722	march	0.00636	look	0.005362
twitter	0.006499	like	0.005915	just	0.004984
join	0.005957	time	0.005492	get	0.004833
scienc	0.005937	day	0.004683	will	0.00465



3.5 LDA with name translated

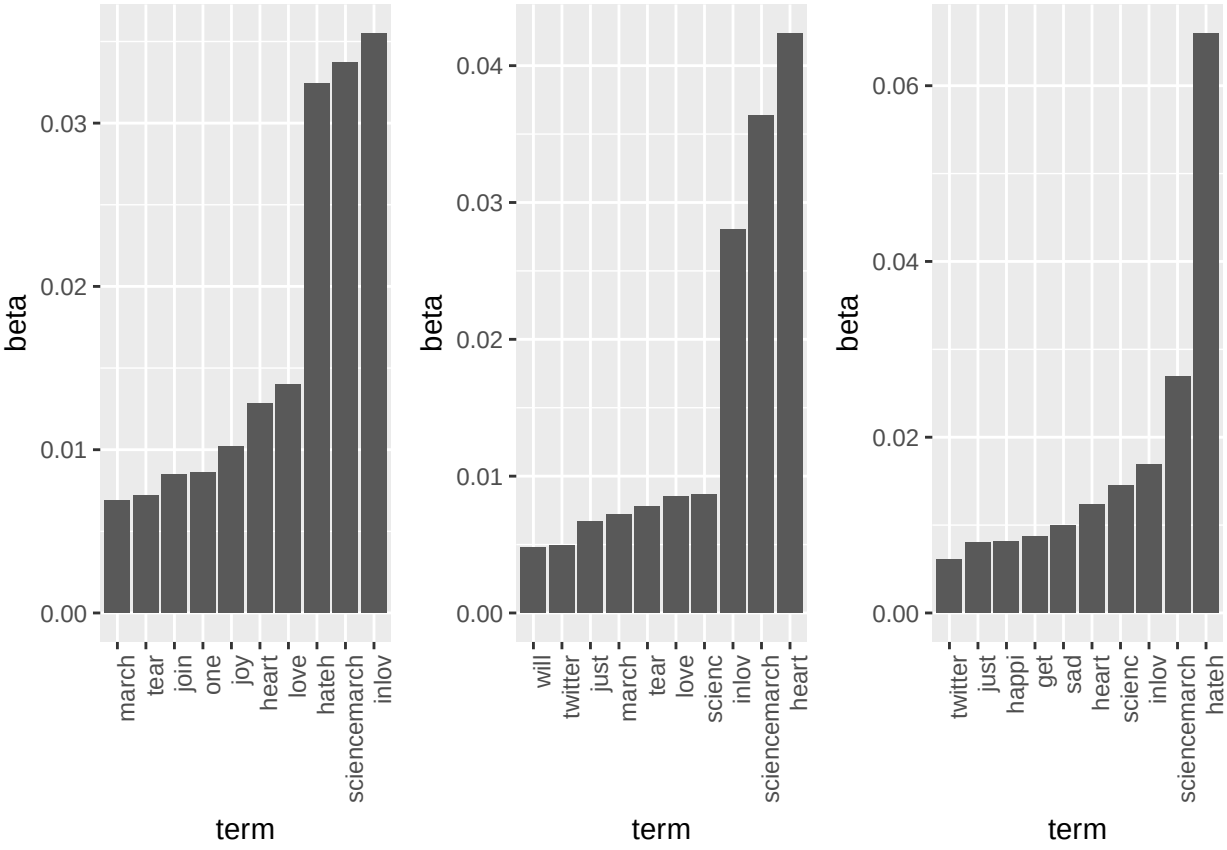
The last case was to perform LDA after traslating the unicode Emoji characters in english. `unicode` package was used to match the unicode to its name. Then the standard process of stemming and deletion of stop words where performed.

Table 7: Output of LDA with translated Unicode

Topic 1	Topic 2	Topic 3
inlov	heart	hateh
sciencemarch	sciencemarch	sciencemarch
hateh	inlov	inlov
love	scienc	scienc
heart	love	heart

Table 8: Word prob. given topic

1.term	1.beta	2.term	2.beta	3.term	3.beta
inlov	0.03549	heart	0.04235	hateh	0.06595
sciencemarch	0.03372	sciencemarch	0.03638	sciencemarch	0.0269
hateh	0.03243	inlov	0.02805	inlov	0.01693
love	0.01397	scienc	0.00868	scienc	0.01448
heart	0.01281	love	0.008492	heart	0.01239
joy	0.01018	tear	0.007807	sad	0.009971
one	0.008586	march	0.007194	get	0.008753
join	0.008463	just	0.006715	happi	0.008093
tear	0.007214	twitter	0.004922	just	0.008068
march	0.006867	will	0.004763	twitter	0.006052



4 Conclusion

As the result of the exploratory analysis indicates, user-generated-contents may contain Unicode Emoji characters. These Emoji characters sometimes carry mixture of condensed information that is difficult to express in words. The result of the output from the LDA indicates that words such as “heart” that would have been neglected using the traditional method may be saved when the Unicode characters are translated into meanings.

5 Appendix

6 `emoji` package in R

6.1 Emoji Data Set

Plan to change this part after posting the Emoji package on CRAN

Emoji in a text data is encoded as a sequence of Unicode: an industrial standard that consists encoding, representation, and text expression of writing system. *The Unicode Standard* is distributed by a non-profit organization the *Unicode Consortium*. The current list of Emoji v5.0 is available on the official *Unicode Consortium* website. Example illustration of the Emoji table is attached in Figure 2. Data set of Emoji characters are available in `emoji` package.

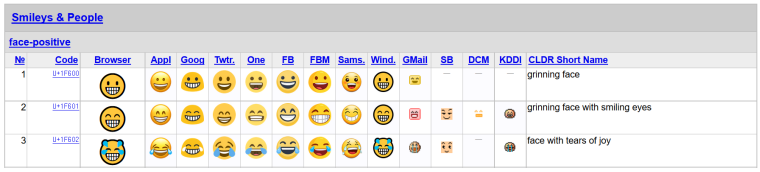


Figure 2: Glimpse of the table of Emoji on the unicode.org website

6.2 Generate different type of encodings using Python

A script was written R that changes between different encoding environment. Plan to include this feature in the 'emoji' package.\ There are multiple way of encoding Emojis on website or SNS. Unicode, Unicode escape, UTF-8hex, zerox notation, NCR are examples of commonly used encodings. As shown in Figure 2, the initial data set scraped from the *Unicode Consortium* only has Unicode. The most common encoding type for online web page, however, is UTF-8 and Unicode escape. Therefore, the original Unicode sequence should be translated into different encoding types for the data set to be applied. Different types of encoding format were generated from the original Unicode via simple Python code.

6.3 Scoring of Sentiment

The characteristic of Emoji (effectively delivers feelings and moods), naturally leads text mining with Emoji to sentiment analysis. `tidytext` package in R has three general purpose lexicon sets. The `AFINN` score words from -5 to 5 scale, `bing` assigns words in binary category(positive and negative), and `nrc`assigns words with more categories.

6.4 Description of the Emoji Data set

The table should be updated. It is using the v4.0 Emoji list

The complete Emoji data set is saved under the 'data' directory. This complete data set is read and named 'uni_info'. Some Emojis are a combination of two or more basic Emojis. For example, Emoji 'boy: light skin tone' is a combination of 'boy' (U+1F466) and 'light skin tone' (U+1F3FB). Data 'basic_uni_info' is a data set of the basic Emojis. There are many different ways to encode 'Unicode'. The data set includes the following encoding types: 'U+hexadecimal', 'UTF-8 hexadecimal', 'hexadecimal', and 'numeric character reference (NCR)'. The example of the data set is given in Table 9

Table 9: Information of 5 Emoji data set

uni_No	uni_code	uni_name	uni_age	uni_keyws	utf_8_hex	zerox_notation	ncr	PosScore	NegScore
1	U+1F600	grinning face	2012	face	f09f9880	0x1f600	128512	0	0
1	U+1F600	grinning face	2012	face	f09f9880	0x1f600	128512	0	0.5
1	U+1F600	grinning face	2012	face	f09f9880	0x1f600	128512	0.125	0.125

uni_No	uni_code	uni_name	uni_age	uni_keyws	utf_8_hex	zerox_notation	ncr	PosScore	NegScore
1	U+1F600	grinning face	2012	face	f09f9880	0x1f600	128512	0.125	0.375
1	U+1F600	grinning face	2012	grin	f09f9880	0x1f600	128512	0	0