音码：Phonetic code
形码：Graphic code
义码：Semantic code

A contrastive study on Phonetic code program and Graphic code program

Liang Jinbao

March 23, 2018

1. Question about how to choose a suitable program.

While making or choosing a Mongolian coding program, two aspects should be considered.

- Advantages and disadvantages of this program itself

Every program has both advantages and disadvantages. Phonetic coding program has its own benefits, so does graphic coding program.

- Social costs of introducing this new program

If we have to choose one system from multiple systems, the advantage of this system which could be the key reason to choose this system is really important. However, considering to switch the current system to another, we should think about the social costs of introducing this new system except its own advantages and disadvantages.

2. Contrasting and analyzing fairness of these two programs

Considering current situation of Mongolian coding, we should admit that phonetic coding does have some issues and most of these issues are not because of the coding program itself. The aim of contrasting is to find the best way to solve those issues. It depends on two aspects about whether final effect is good or bad:

- Inherent pitfalls of the program itself
- Way and method of implementing the program

Only when we contrasted the inherent pitfalls of the program itself, could we achieve the goal of comparing. Otherwise we may confused by the current situation of the phonetic coding program and then is lured to change this situation radically.

In order to develop the current phonetic coding program and achieve a stabilized and unified coding system, we made a modification of the current phonetic coding system. Details see another document “Revised Proposal for minimizing the current phonetic code. docx” .

3. Diversity of the needs of the society

There is no perfect program in the word. The solution program is in a complicated environment which multiple needs exist all at one. It may be perfect in solving one kind of issue while showing its disadvantages in solving another kind of problem. For instance:

	词码到词义 Code2mean	词形到词码 Shape2code	搜索 Search	排序 Sort	输入法 IME	用户直觉 sensible intuition	字体实现 Font Implement
Phonetic	★			★	★	★	
Graphetic		★	★				★

* Asterisk means that there is an advantage. 2 stands for to

Different sections have difference needs, so we shouldn't value good/bad according to the number of stars. This graph is only pictured to explain that there are many reasons to influence good/bad of one program. What's more, with all these developing the technologies, disadvantages of every program could be made up by assistant tools and measures such as engine (graphic advantage), sequence (phonetic advantage), input (phonetic advantage) and etc. In conclusion we should serious about pursuing a pure theoretical perfect program.

4. Fonts display

It is no doubt that graphic coding program is convenient in fonts making. It is also true that the current phonetic coding program unifies in coding but incompatibility problem still exists in fonts making. There are two reasons that causes this phenomenon: inherent pitfalls of the program itself and way and method of implementing the program.

(1) Model of the current phonetic coding program is very complicated and detail specification defects.

Model indeed is complicated. If we only consider from the angle of variant rules, we could know that Mongolian variant rule itself is more complicated than Arabic. The current phonetic coding program is lack of unified specific variant norms and has a few versions of variant rules that incompatible. However we should know that this is not because of the program itself but because of not establishing relevant national norms in time.

In order to solve this issue, we have been trying to establish phonetic coding norms and we had already set a program to unify and make standard variant rules. A good result was achieved recently. Unifying has been achieved in a few big information companies like Menksoft, Delihai, Galatu and etc. In the end of this year nearly all Mongolian internet would update fonts and input that support this program standard.

(2) The current phonetic coding program could cause ambiguous hypertext makeup easily.

Compared with graphic coding program, one disadvantage of phonetic coding program is different codes of the same graphic alphabet. There are a few vowels that cause textual ambiguity:

A. Mixing vowel O/U/OE/UE because of dialect differences.

It accounts for the largest proportion and it comes from places where Mongolian spread and it has dialect differences.

B. Maliciously make up fonts (alphabet & FVS)

It accounts for a small proportion, it could happen whether one knows or doesn't know Mongolian and it generally appears in fields that has printing requirements and displaying requirements.

C. Input word according to the picture

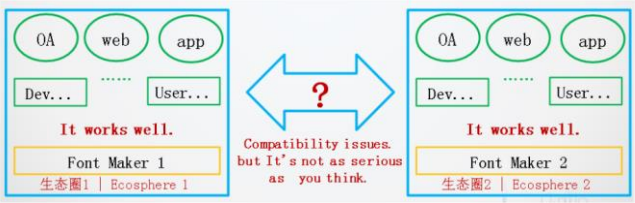
It accounts for a small proportion and people who don't know Mongolian and could see the words appearances and then input words.

D. Unforeseeable extra FVS interference

It accounts for a small proportion. Human error makes unforeseeable extra FVS interference.

How serious this problem is? If just considering from researchers' angle, we think that the deeper we searched, the more serious this problem is. In fact problems have been existed in the actual social environment, and it is not very clear or we should say it has not quite reached in the situation that coding program has to be changed. The negative social impact of this program is less than changing a new program.

The following picture is about current ecosystem of Mongolian information product.



Windows and internet page system that could setup the third party fonts have many Mongolian information products which could satisfy almost every field needs, even the field that common people don't really care about like News, TV, publishing, education and etc. There is no barrier to use Mongolian on computers in work, study and daily life. Only if all products in the solution used the products of the same ecosystem, could we have a stable program and no serious incompatibility problem happens. Some small issues like wrong spell could be solved by assistance tools. It means that incompatibility problem still exists in current program and we are not satisfied about it, but I still want to confirm that the current phonetic coding program is not as bad as expected.

Clearly, we are hoping to change the situation in this meeting that there are still computer systems like iOS and Android that users couldn't setup the third party fonts on it.

5. Input method

Mongolian is alphabetic writing. There are two kind of inputs recently: alphabets-input method and words-input method. No matter which one, users have to calculate letter structure of the word before they try to input it and then use the letters mapping keyboard to type in the Mongolian word. If touched the keyboard that sequence many words, users need to use free variant choosing and controlling character and also they could use the words-input method which wouldn't need to type free variant choosing and controlling character.

Based on analyzing the above Mongolian input logic and progress, we could know that the current phonetic coding program has natural advantage in many steps input method, re-editing and etc. As for graphic coding program, though we could use input method like Wubi input method, still users can't do without words-input method.

6. Sequence

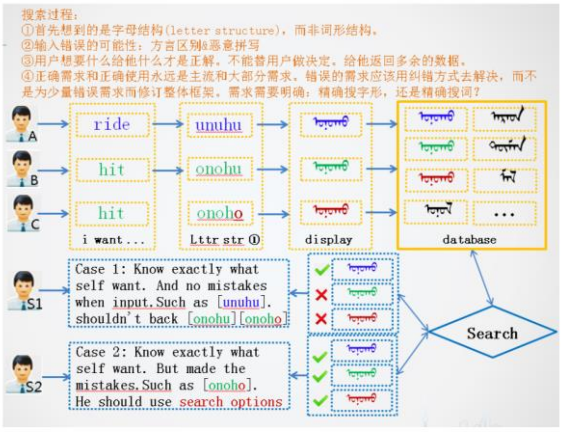
The current phonetic coding program is developed based on Mongolian alphabet structure and it is convenient naturally in sequence. The only aspect that we should consider is free variant choosing and controlling character in the word. In this character, graphic coding program need to do a more complicated calculation to reach the effect of phonetic coding program.

Here I should clear that whichever program could make up its own disadvantage by using assistant tool and could reach to the nature advantage of another program.

7. Searching

Like every program has its advantage and disadvantage, searching is a very tiny small question that exists in the current phonetic coding program. It is the same as graphic coding program who has sequencing complicated problem in order to pursue unifying of word and structure. If only focusing on word structure, graphic coding program is the perfect one. However, Mongolian users (especially while dealing with natural language) never focus on word structure, instead they focus on letter structure. The differences between Mongolian textual language (letter structure) and spoken language (pronunciation) are large. In result, Letter structure of words is the most important thing in Mongolian. In actual requirement, specific letter structure searching is the real request instead of vague graphic searching (vague means same structure while different codes).

Compared with graphic coding program, the current phonetic coding program is a double-edged sword in the issue of searching. Graphic coding program could not precisely search l letter structure while phonetic coding program not only could easily do vague graphic searching with the help of assistant tool, but it also could precisely do letter structure.



Searching progress:

①: Thinking about letter structure first instead of thinking word structure.

② possibilities of wrong input: Different dialects or malicious spelling.

③ Giving exactly whatever users need and do not make decisions for them and do not give them extra data.

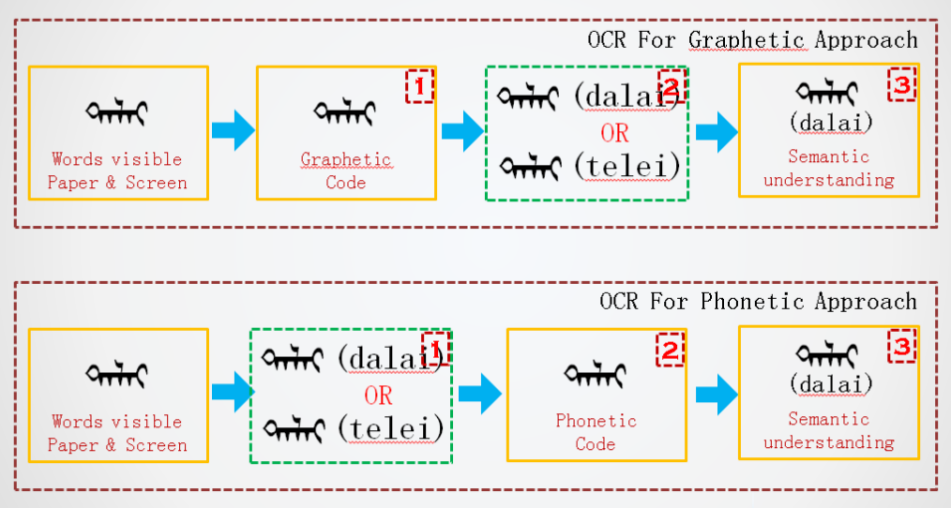
④ Correct requirement and correct using are always the mainstream and the most request of the users. Instead of changing the whole frame of the program because of one tiny issue, wrong request should be solved by error correction

methods. Request should be precise: specifically search letter structure of word or the word itself.

8. Shape to Code

The starting point of this application situation is the visible text requirement. The typical case is OCR. If the aim of this application is to switch the visible word structure to coding (it is graphic coding in this case), graphic coding program is much better than phonetic coding program.

Here is how it works:



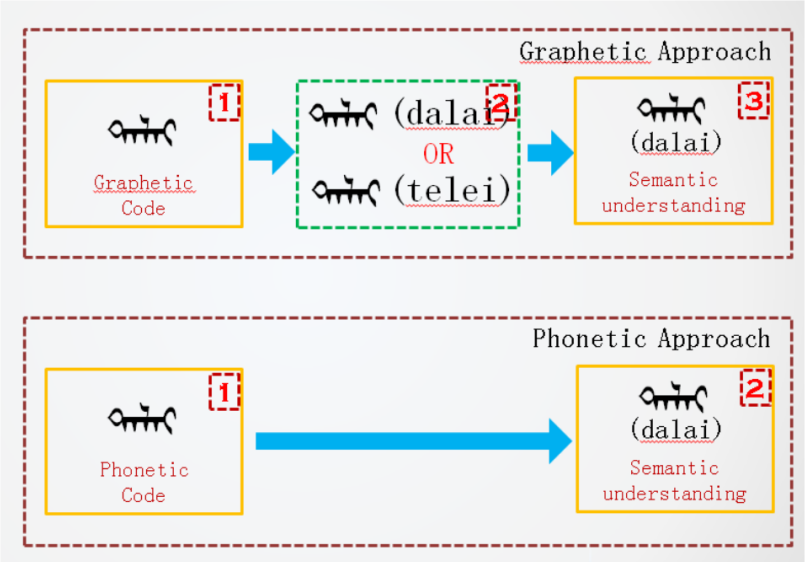
In graphic approach, the progress is simple from original picture (visible word picture) to the aiming code (graphic coding) and there is no ambiguity. As the first step showed on the picture, mapping directly from shape to code one by one. However, the finial aim of the computer is not simply calculating code. It has to deal with semantic calculation as the step 2 showed in the picture. In this step, ambiguity issue would appear. In order to precisely calculate semantics, we must deal with ambiguity issue.

In phonetic approach, it is more complicated from original picture (visible word picture) to the aiming code (phonetic coding). We should do complicated calculation by using ambiguity (step 1) and then typing out the actual code (phonetic coding). However the forward step 3 is much easier. Ambiguity disappears while doing semantic calculation.

In conclusion, comparing graphic coding program and phonetic coding program in character recognition field, we could know that graphic coding is not better than phonetic coding. It only have advantages if the target task is to recognize coding. If the target task is to recognize semantics, its practical computer burden is the same as the phonetic coding program.

9. Coding to Meaning

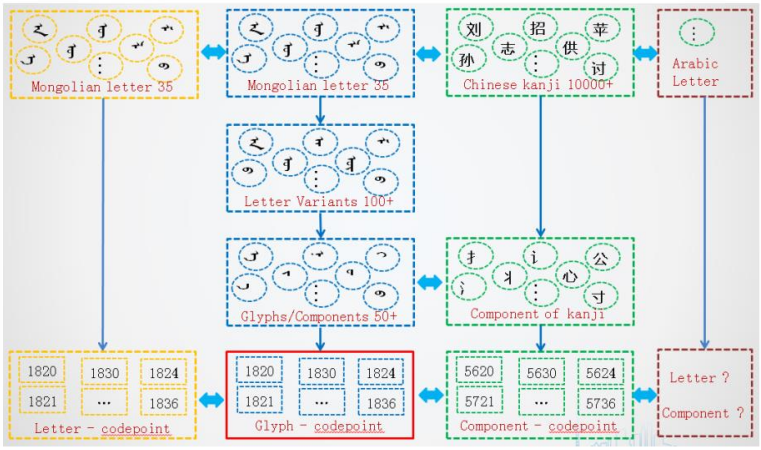
This kind of application situation is the progress to calculate invisible coding to meaning. The superiority of phonetic coding program is quite obvious because phonetic coding itself has the function of letter structure. We could easily see through the picture followed that graphic coding program has increased the possibilities of same structure different meanings while under the case of decreasing possibilities of same structure different codes.



10. Interpretation of graphic coding program

Graphic coding program tend to focus on graph and decrease ambiguity between graph and codes. However ambiguity of graph and letter structure has been increased. Phonetic coding program tend to focus on alphabets and decrease ambiguity between codes and letter structures, however ambiguity of code and graph has been increased.

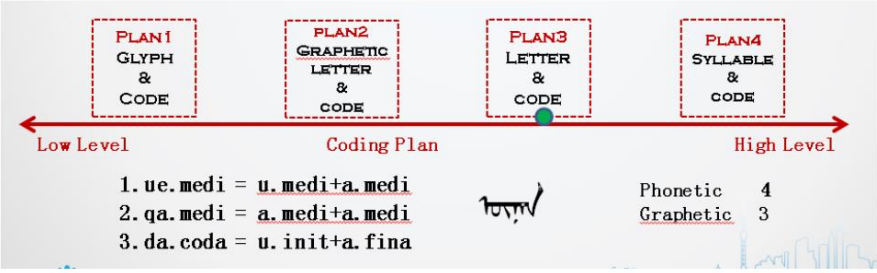
In result, everyone was trying to avoid ambiguity which is not from the coding programs themselves but from design of this character itself. In order to solve this issue, graphic coding program divided Mongolian letters and their variants freely according to unit type and then formed a system of virtual graphic letters and then coded this system. Here is how it works:



About Graphic coding, two more questions should be considered:

1) Did graphic coding program really solve ambiguity between graph and code?

As some letters still maintain part of Mongolian characteristics while making this virtual graphic letters system, ambiguity problem still exist and disadvantages like ambiguity and safety still exist as the current phonetic coding program.

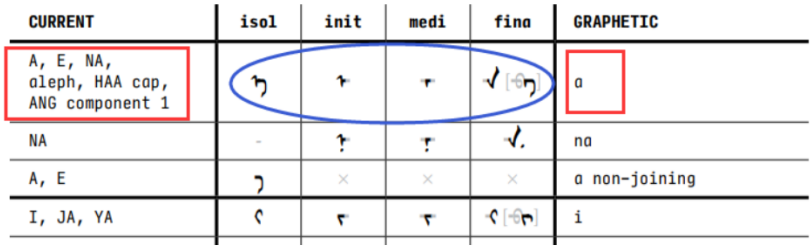


2) Why is it not easy to accept graphic coding program emotionally?

Arabic and Mongolian are both alphabet writing. However, Mongolian variant rules are more complicated than Arabic. E.g.:

- Mongolian in-word positional variants is not only one while Arabic has only one variant in one position.
- While varied Mongolian words according to Mongolian orthography, it is not only depends on in-word position but also depends on context grammar while Arabic wouldn't need to consider context. For now, Arabic Ligature model without using FVS only fit for the character system who has only one variant in one position. In order to solve this issue, we have to update ligature model to make it qualified for more complicated language or we have to simplify Mongolian variant rules. Clearly, in order to decrease the Mongolian variant number into one, we have to divide Mongolian letter variants according to font component and re-edit a new graphic writing system which is neither letter nor letter variant.

Here is how it works:



CURRENT	isol	init	medi	fina	GRAPHETIC
A, E, NA, aleph, HAA cap, ANG component 1	ᠠ	ᠡ	᠊ᠠ	᠋ᠠ	a
NA	ᠨ	ᠨᠠ	᠊ᠨᠠ	᠋ᠨ	na
A, E	ᠡ	×	×	×	a non-joining
I, JA, YA	ᠢ	ᠢ	᠊ᠢ	᠋ᠢ	i

In the above picture, we could see that a graphic letter could be a vowel or consonant linguistically. This kind of multi-vowel/consonant indistinctive coding program has already made graphic letter system losing many Mongolian characteristics including Mongolian essence vowels, consonants and etc.

It is hard to decide positive/negative of words, do Latin transliteration, syllabify, calculate vowels and consonants and etc. Those are all characteristics and advantages of alphabet writing. We shouldn't have to deprive special cultural connotation of languages for the purpose of convenience and usability.

It is easy for people to feel that we shouldn't destroy a language and cultural just for a technic issue for it may change people's mind if we deprive special cultural connotation of languages for the purpose of convenience and usability. It is only sensible disagree except rational analyzing advantage/disadvantage of a technology.

3) Big companies are waiting for a stable coding program

We shouldn't have the idea that only simple and understandable graphic coding program is a stable version. Standardized and unified phonetic coding program also is a stable version.