Subject: Re: String near match algorithm
Posted by Didier on Mon, 28 Dec 2009 11:12:24 GMT
View Forum Message <> Reply to Message

Mindtraveller wrote on Sun, 27 December 2009 13:25OK. If I search for a word in text, I should split text into words and apply near search algorithm to each word.
1. Is it right?
2. Which value should I compare function result to in each case?

Hello Mindtraveller and Koldo,

My small algoritm can compare whatever you like if you modify it a bit. But it is originally intended for string comparison.

and it compares the complete texts ==> this means that if you want to find a near match inside a phrase you will have to compare all the words individually
===> 1: YES

===> 2: The following function is what I use to determine if it is a near match or not.

```
inline bool CompareDistance(const String& a, const String& b)
{
 if (correlation(a, b) >= max(2, min(a.GetLength(), b.GetLength())*3/5)) return true;
 return false;
}
```

The (3/5) value is a threshold value that you can tune to your needs but this one works pretty well. The max() and min() functions are to treat corner cases where the words become very small, in fact it is directly linked to the following code in the correlation() function: int matchPatternMinLength = max(2, min(b.GetLength(), a.GetLength())/3)'

I'm gonna make a zipped project with all in it.