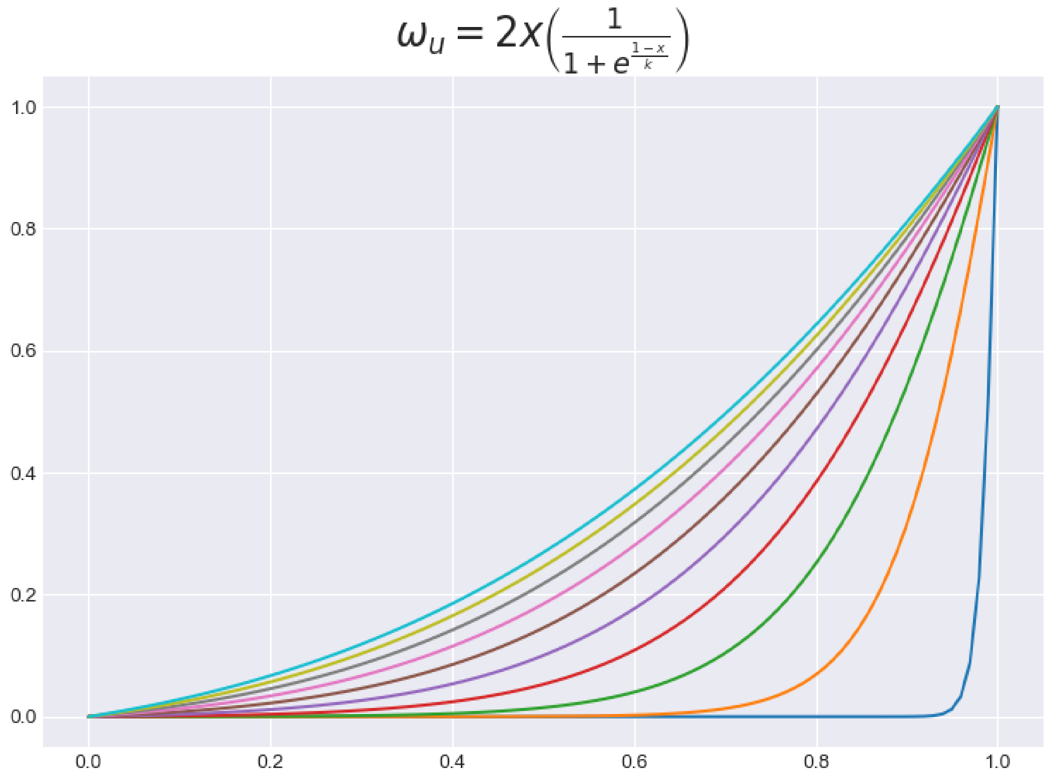
Dynamic weight generation for vehicle signal set matching.

Peter Winzell, Volvo Cars, Sunnyvale, Lund



Abstract

The problem of matching vehicle signal sets and perform translation and mapping from one signal set A to another signal set B is currently a manual task that is both time consuming and tedious. Furthermore, translation to an open standard such as VSS [1] - Vehicle Signal Specification – promotes innovation across the industry and it also automatically hides proprietary signal details. The proposal of a ranking algorithm [2] where the attributes associated with a certain signal are compared to the same attributes with a signal from another set can be used to facilitate signal set mapping. The algorithm returns a score that is used to estimate how well two signals from different sets match. To calculate the score, we have a score and a weight for each attribute comparison. Typical signal attributes are name, description, unit, and datatype. The weight is used to value attributes against each other. The comparison of one type of attribute is worth more to the overall signal similarity than others. i.e., name and description are worth more than unit and data type. For this purpose, we introduce a dynamic attribute weight function that reflects the value of a comparison of greater

importance. We define a dynamic weight function that is increasing in the interval [0,1]. We show that we can control the weight and its significance for the overall similarity of two vehicle signals from different vehicle signal sets. This will ultimately lead to an improved way of performing signal set matching in-vehicle as well as for recorded data sets in the cloud.

Introduction

A ranking algorithm that generates a score based on how well two signals match can be described as follow:

$$p = \frac{\sum_{i=1}^{n} f_i w_i}{n}, where\ f_i \in [0,1] and\ w_i \in [0,1] ,$$

$$f_i\ is\ the\ comparision\ function, w_i\ is\ the\ adjusted\ weight.$$

$$n\ is\ the\ number\ of\ attribute\ matching\ functions.$$

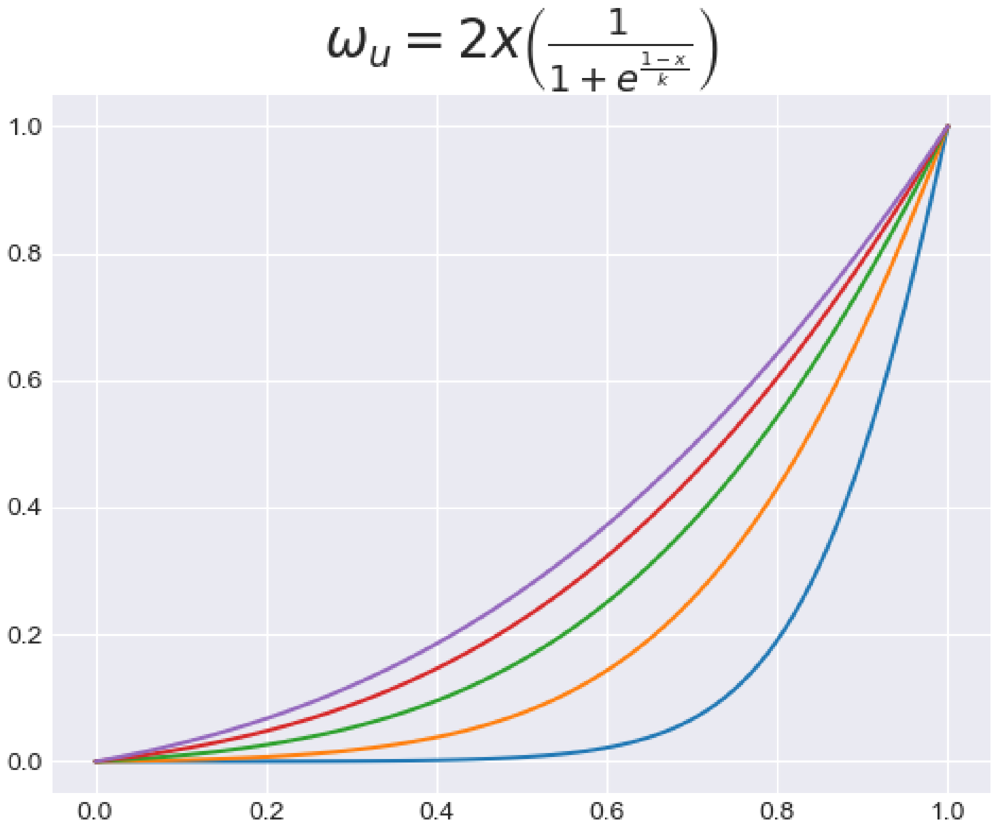
Now, if we look at a vehicle signal attribute unit which can be m/s, km/h, kg etc. these units can be associated with different signals from one specific signal set. For example, a sensor measuring wind speed and another measuring the vehicle speed.

This means that we would like to adjust the ranking algorithm so that unit comparison is weighed less than for example the textual description attribute. Having just a static weight associated with an attribute we end up with an equal score for units that are exact matches, even though the signals are disjunct [2] . To mitigate this, we introduce a weight attribute generator f_w that takes the value from what we consider a more important attribute function to generate a weight $w_i \in [0..1]$, a weight that is then used to calculate the score of a less important attribute. We also are defining the function so that we can control when we want an increase of the weight w_i as the more important attribute score approaches 1.

Function definition

We need an increasing weight function f_w, *where* $0 < f_w < 1$.

$\boldsymbol{f_1}$:



We define our weight function as:

$$f_w \; = \; g(f_i \cdot w_i) \; = \; 2x\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right), \; \mathrm{x} = f_i \cdot w_i, \;\; 0 < x \; < 1, \;\; k \in [0.01{,}0.5]$$

The selection of constant k lets us define where we want the weight function to approach 1 rapidly. This way we have invented a dynamic vehicle signal weight attribute generation where we can dynamically control the weight, i.e., how important we believe the attribute comparison has become from our starting signal attribute.

We have

$$\lim_{x \to 1} f_w \; = \; 2x\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right) = 1$$

$$\lim_{x\to 0} f_w = 2x\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right) = 0$$

We now need to prove that f_w is an increasing function in the interval $[0,1]$.

We have that $f_w = g(x)\cdot h(x) = 2x\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right), g(x) = 2x \text{ and } h(x) = \left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right)$.

We only need to show that the derivate is positive within the interval, from that it follows that f_w is increasing within the interval.

$$f'(x) = g'(x)\cdot h(x) + g(x)\cdot h'(x) = \frac{dy}{dx}(2x)\cdot\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right) + 2x\cdot\frac{dy}{dx}\left(\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right)\right) \Longrightarrow$$

$g'(x) = 2,$

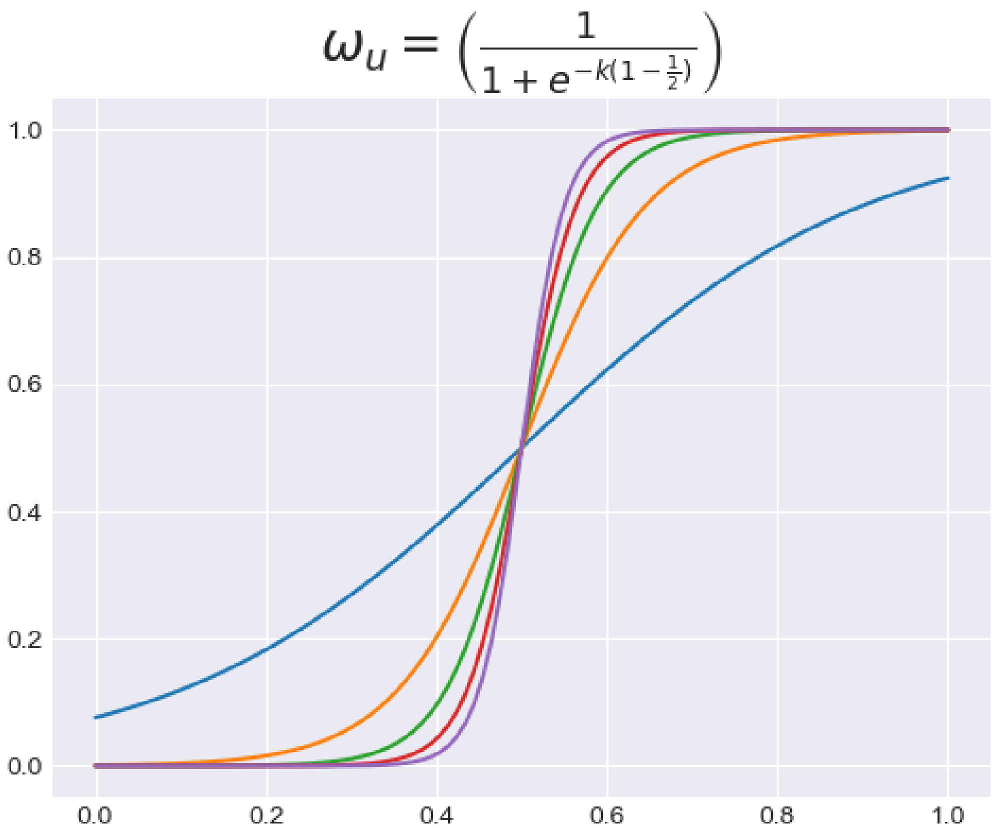
$$h'(x) = \frac{e^{\left(\frac{1-x}{k}\right)}}{k\cdot\left(1+e^{\left(\frac{1-x}{k}\right)}\right)^2}, \Longrightarrow$$

$$f' = 2\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}} + \frac{xe^{\left(\frac{1-x}{k}\right)}}{k\cdot\left(1+e^{\left(\frac{1-x}{k}\right)}\right)^2}\right)$$

$$f'(0) = 2\cdot\left(\frac{1}{1+e^{\left(\frac{1}{k}\right)}} + \frac{0\cdot e^{\frac{1}{k}}}{k\cdot\left(1+e^{\frac{1}{k}}\right)^2}\right) = 2\cdot\left(\frac{1}{1+e^{\left(\frac{1}{k}\right)}} + 0\right) > 0, \forall\, k \in [0.01, 0.5]$$

$$f'(1) = 2\cdot\left(\frac{1}{1+e^{(0)}} + \frac{1\cdot e^0}{k\cdot(1+e^0)^2}\right) = 2\cdot\left(\frac{1}{2} + \frac{1}{4k}\right) > 0, \forall\, k \in [0.01, 0.5]$$

f_2:



A slightly different function is also being investigated:

$f(x) = \frac{1}{1+e^{-k\left(x-\frac{1}{2}\right)}},\ 0 \leq x \ \leq 1, k \in [1,5,10,40]$

As above we define our weight function:

$f_w = \ g(f_i \cdot w_i), where\ f_i\ and\ w_i$
$is\ the\ weight\ and\ calculated\ comparision\ value, i.e\ the\ description\ function\ value$
$and\ its\ weight\ in\ our\ exqmple.$
We set $x = \ f_i \cdot w_i$.

$$\lim_{x\to 1} f(x) = \frac{1}{1+e^{-k\left(\frac{1}{2}\right)}} \ = \ 1\ when\ k > \sim 10$$

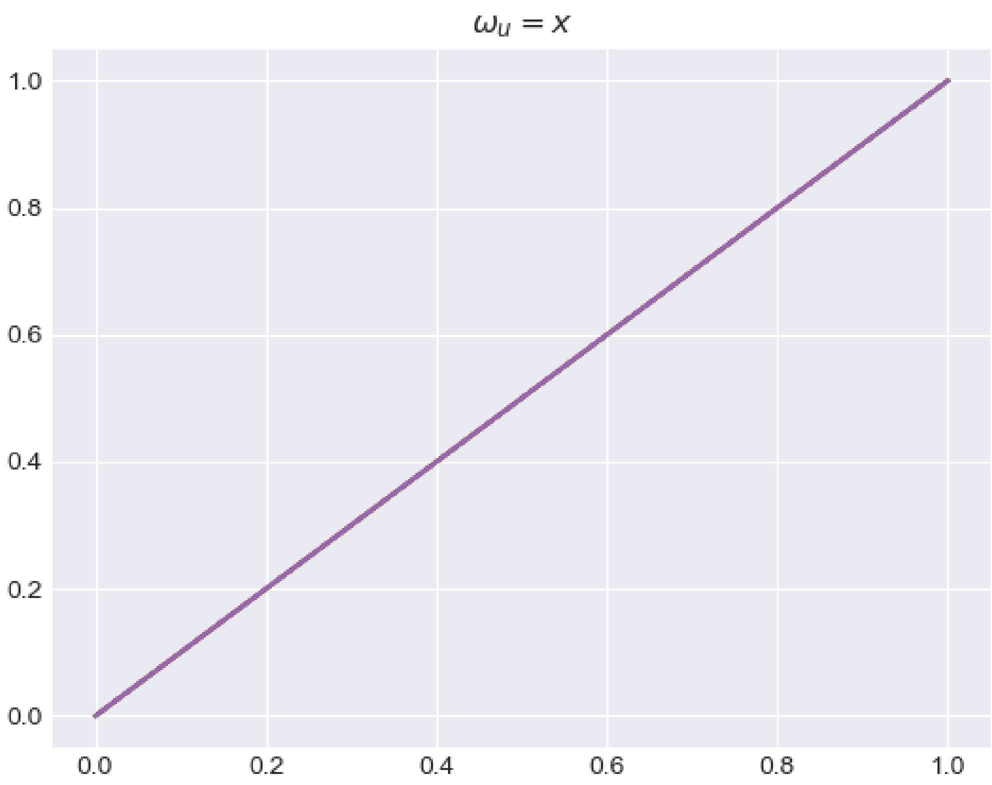
$$\lim_{x\to 0} f(x) = \frac{1}{1+e^{-k\left(\frac{1}{2}\right)}} = \ 0\ when\ k > \sim 10$$

$$\lim_{x\to\frac{1}{2}} f(x) = \frac{1}{1+e^{0}} = \frac{1}{2}$$

$$f'(x) = \frac{k \cdot e^{-k(x-\frac{1}{2})}}{\left(1+ e^{-k\left(x-\frac{1}{2}\right)}\right)^2} > 0 \; \forall \; x \in [0..1] \; k > \sim 10$$

The function is increasing when we select k as above.

$\boldsymbol{f_3}$:



We also test a simpler function:

$$f(x) = k \cdot x, where\; k = 1 \; and \; x \, \epsilon [0..1]$$

$$f'(x) = 1 > 0 \; \forall \; x \in [0..1]$$

The match delta as a function performance criterion

We define a criterion to be able to compare how well the weight generating functions perform. We measure the delta between matches and no matches. The performance criterion should reflect that we want matches to generate a high value for dynamic weights and a low value for no-matches.

We define P_c as:

$\Delta_{match} + \Delta_{no-match} = P_c$
$\Delta_{match} = f_{dyn} - f_{static}$
$\Delta_{no-match} = f_{static} - f_{dyn}$

Example

Let's look at some examples where we compare a small subset of signals from two signal sets A and B.

A:

Name	Vehicle.Engine.Speed
Description	Engine speed measured as rotations per minute
unit	rpm

B:

Name	WinWipgAutCmdIf
Description	Distributes information about wiping speed in rpm
Unit	rpm
Name	EngSpdDisp
Description	Engine speed value for engine speed meter
Unit	rpm

So, we have two signals in B that share the same unit, but the actual signals are disjunct. The attribute functions for description and unit for these signals are defined as:

$let\ p_d = f_d \cdot w_d\ be\ the\ description\ attribute\ similarity\ score,$
$and\ p_u = f_u \cdot w_u\ be\ the\ unit\ attribute\ similarity\ score \implies p = \frac{p_d + p_u}{2}.$

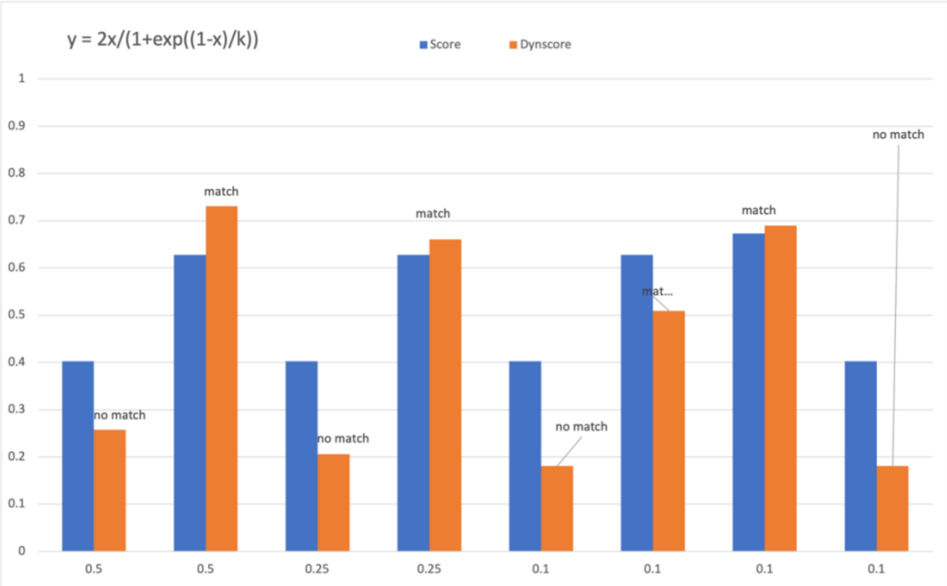
We use the following 3 functions:

$$f_1 = 2x\left(\frac{1}{1+e^{\left(\frac{1-x}{k}\right)}}\right)$$
$$f_2 = \frac{1}{1 + e^{-k\left(x-\frac{1}{2}\right)}}$$
$$f_3 = kx$$

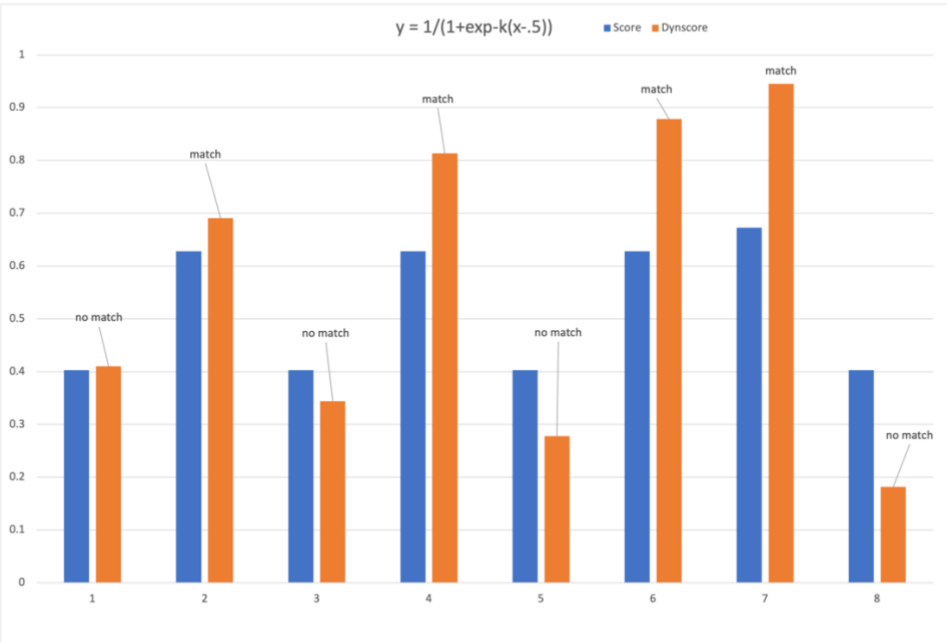
The diagrams below show comparison between signals that should match and signals that should not. It also shows the algorithm using static weight and with dynamic generated weights. The diagram indicates that the dynamic weight generation improves the ability to match signals from the signal sets.

The constant k selection is tailored for each function respectively. What we are aiming for is our score to increase when similarity score is approaching 1 and decrease when we are closer to 0.

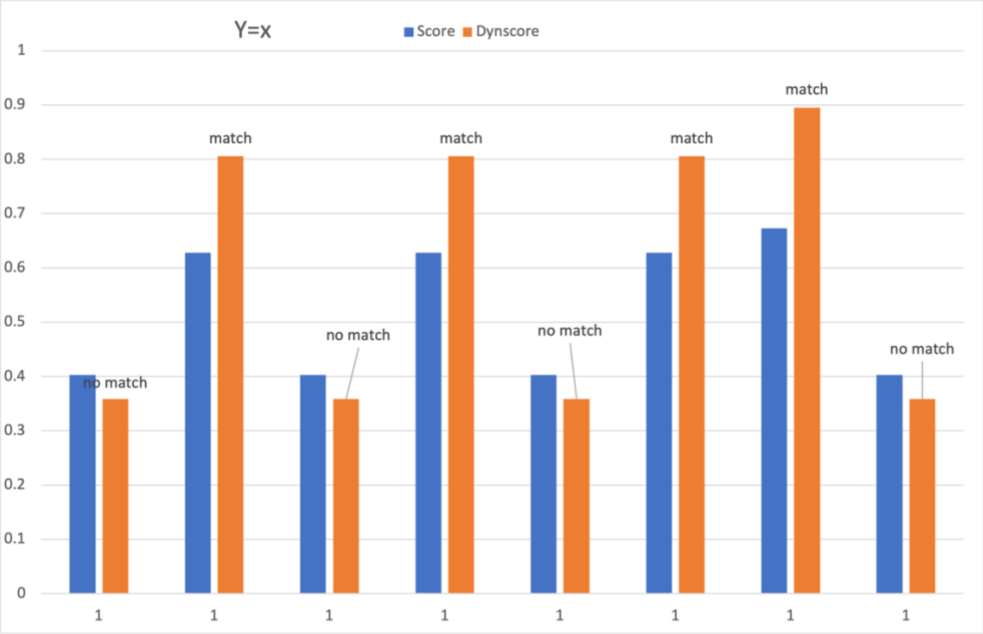
f_1:



f_2:



f_3:



Using the match delta as a mean to estimate the performance we get the following results:

$P_c^{f_1} = \Delta_{match} + \Delta_{no-match} = 0.82$
$P_c^{f_2} = \Delta_{match} + \Delta_{no-match} = 1.17$
$P_c^{f_3} = \Delta_{match} + \Delta_{no-match} = 0.94$

We can also look at the specific selection of k, whereas we need to keep k constant for f_3 we get the same score, the f_1 *and* f_2 results will vary depending on the selection of k. If we look at the top delta we get:

$f_1^{k=0.5} = 0.25$
$f_2^{k=40} = 0.5$
$f_3^{k=1} = 0.27$

Selectin' the slope pace, $f(x) = \frac{h(x)}{e^{k \cdot g(x)}}$

Selecting the value of k for these types of exponential functions is important for the performance of the algorithm – when do we think we have a match, or a no-match. Calculating the delta between a match and a no-match we can verify the algorithm. We can try to search for an optimal or near optimal value of k using reinforcement learning or also possibly by using a genetic algorithm. It would also be possible to use multiple functions within the interval:

$$\text{f(x)} = \begin{cases} g(x), 0 \leq x \leq \alpha \\ \quad\quad \ldots \\ h(x), \gamma \leq x \leq 1 \end{cases}, \{x \in \mathbb{R}:\ 0 \leq x \leq 1\}$$

In the end the overall outcome and accuracy of the algorithm depends on how well we are able to tune the language models that compares the description text from two separate vehicle signal descriptions.

Works Cited

[1] COVESA, «VSS,» COVESA Alliance, 2023. [En ligne]. Available: https://covesa.github.io/vehicle_signal_specification/.

[2] J. S. Peter Winzell, "Ideas for signal set matching," Volvo Cars, October 2022. [Online]. Available:https://wiki.covesa.global/download/attachments/34209833/SignalSetMatching.pdf