The Application

I would like to start with the Application, as to which there was some confusion yesterday, but I think we got there in the end.

The Judge summarises in [4] – [13] of the Judgment, and which we summarised in Annex 2 to our High Court skeleton – which is in [SUPP/1/p/41]

If I may start in the **Judgment at [8] and [9]**.
There the Judge sets out the two ANNs which are involved in the *training* process.

The first, which he **discusses in [8]**, is the so-called semantic analysis. This is a process used only in training, and what it involves is taking a written, hence the word semantic, description of a track and running it through an ANN which can analyse text, for embedding the track in the semantic embedding space.

Appn 7$_{20}$ - 8$_{12}$,

Pardoe §50 - 52

Text analysis of this type was established at the priority date, and this analysis could be done with an off the shelf NLP text analyser, which is a kind of ANN.

This produces a set of coordinates, that is a vector, which points to a particular location in the semantic embedding space.

Answering ML LJ Birss' question of yesterday, our understanding is that it is a vector in the sense that it describes the movement necessary from the 0,0,0 origin.

Note that NLP simply looks at the words use and perhaps their relationships in the text in a purely mechanical way. Whilst it manages to group text with similar meaning, it does not do so by any understanding of that meaning, and it does not work in that way at all. It is a mechanical analysis with no intelligence or understanding behind it.

For that reason a human cannot 'read' that NLP vector and understand what it is indicating – that for example a particular vector means that the description is of a "happy, light track with a summery feel" or whatever.

The embedding space does not work in that way, and that the vector cannot be read off by a human like that.

But what one does know is that whatever the semantic meaning of the text associated with a first track, another track whose semantic vector is closely associated with it will have a similar semantic description; whatever that description might be.

And conversely one located distally to the first track will have a different semantic description; again without any particular knowledge from their vectors of what that meaning might actually be.

Thus the **scalar** distance between two tracks is used in the invention as a measure of semantic similarity or dissimilarity; but without any understanding of their actual semantic nature.

Secondly, as **the Judge sets out in [9]**, tracks are analysed by a different, separate ANN for their **physical** properties, for embedding in the physical properties embedding space. This is a purely physical analysis, for music based on for example beats per minute, also tonality, rhythm, timbre, based for example on a thing called a Mel spectrogram.

App^{n} 23$_{27}$ – 31$_{6}$

36$_{16 – 27}$

Fig 4

Our skelly §101

Note that the properties themselves are extracted using standard techniques known at the time, and it is those which are fed into the physical ANN.

The physical ANN then processes those inputs to produce a final output vector, which is the one referred to as V_{FILE}.

But again, that vector has no specific or readily human identifiable meaning.

Although one knows that it is based on an analysis of the input physical features, one cannot 'read' the vector and understand anything about it.

That because there is no obvious relationship between the inputs and the output, and as you may have picked up when MLF went to Fig 7 of the Application yesterday, there is no reason to think that even there will be the same **number** of dimensions in the input and output of the ANN.

App^{n} p.131

And you can also see that from the structure in Fig 6 [**SUPP/7/p.131**] because you can see there that all the say 'tonality' inputs feed into every node on the layer below, and so on, down to the output. So there is no 'siloing' of the inputs to particular outputs. So if you say had 10 inputs about tonality, 10 about rhythm, 10 about timbre, etc, you might have, say, 20 or 50 outputs, and none of them would be a 'tonality' output as such. The outputs are each individually a kind of blended consequence of the total inputs (albeit each individual differently blended).

As the Judge notes at the end of [9], this second ANN is the **only** one which is used once trained.

Judg^{t} [9]

In training the two tracks in the first pair are run through this second physical ANN, and two vectors are spat out.

The scalar distance between those two tracks in the physical embedding space is calculated.

There then follows what the Judge described as a significant 'trick' of the invention, which he sets out in [10] & [11], and if I could invite you to read that.

App^{n} 7$_{20}$ - 8$_{12}$,

As the Judge sets out in [12], once trained, the result is that – when faced with new tracks for which there is no semantic description – the ANN can analyse their **physical** properties, find where they lie in the **physical** embedding space, and know that other tracks close to them in that **physical** embedding space will <u>in fact</u> have similar **musical** properties.

Pardoe §50
Pardoe §52
Pardoe §57

Thus by its analysis of the **physical** properties of a track, the ANN can recommend other tracks which are at **musically** similar, even if they are completely different in style or genre or date.

Pardoe §57

And conversely, just through the analysis of physical features the ANN will avoid recommending tracks which are dissimilar in their semantic content, even if they are similar in their style, genre or date.

But there are **four** further points to note here.

First – the process which has been described does not converge the physical and semantic *spaces* themselves. The physical space does not take on the same form as the semantic space – and you can see this most easily from the fact that they may well, and likely will, have a different number of dimensions, i.e., the respective vectors may have different lengths of 512 bits and 1024 bits.

What has been used is the **scalar** (that is directionless) distances in the semantic space to train the physical ANN, but the two spaces do not become the same.

Second – I hope that this will be completely apparent, but the ANN is not making any musical or taste-based judgment. The analysis is based only on the physical properties, and indeed as ML LJ Birss noted yesterday, even the semantic analysis is just a comparison of text *qua* words – there is no understanding of their meaning in the NLP process – nor anywhere else in the process.

The ANN neither understands nor expresses an opinion on the music itself.

Third – the resulting ANN of this training process, indeed the ANN of any training process, will have a structure which is different from any other ANN. A trained ANN is a new machine with a new structure.

Finally, fourth as is obvious, in an ANN the human (I avoid any contentious word) not only does not *prescribe* how it is that the ANN analyses the physical characteristics so that musically similar tracks are closely located in property space, but that is something which the ANN has to determine for itself from the data.

And MLady&MLs will understand the reason for that – it is not known to a human how the physical sounds of a song actually relate to its semantic nature. No one, at least as far as we are aware, has ever worked out how to do that, but by using the invention a machine can be produced which is able to do that task.

And that is an important point – no programmer is able to instruct the machine what to do, since no programmer knows the answer. Conceptually, the "programmer" doesn't even know how to start.

This is what we refer to as the intractable problem in our skeleton, and it highlights one difference between an ANN and a common parlance computer program.

The contribution and the benefits of the invention
MLF has shown you the claims, and I am not going to spend time on them now.

The Judge set out the contribution in [24], citing the Hearing Officer's [53], but if I may go to the Hearing Officer's decision to pick up also on where he set out the benefits of the claimed invention.

HO Decision §§49 and 54

HO Decision §§53 & 54 and then back to end §49 at [SUPP/6/p.117]

Drawing these together, **six** points:

> **First**, the claimed invention is to an ANN-based system which encompasses the ANN but is more extensive than that.
> **Second**, the ANN, as is conventional, may be hardware or software *implemented*. But that is a matter of its implementation.
> **Third**, the system is one which in use receives a track (or other media file) and performs an analysis of the **physical** properties of the track;
> **Fourth**, the ANN is, from that analysis of those **physical** properties, able to identify other tracks which will have similar **physical** properties, but which will therefore have the benefit of being similar musically to the initial track.
> **Fifth,** a message containing the track recommendation and the recommended file is then sent to a remote device.
> **And sixth,** the system has real benefits: namely increased simplicity, speed and accuracy. As a result the recommendation message is quicker, better, and therefore different from the message of the prior art and therefore new.

Decision §§49, 54

One could summarise that even more simply by saying that the invention is a better system for analysing the physical properties of a music track ... faster, more accurately and more simply ... and sending a better recommendation message – the message itself being new, faster and better in the sense of it contains a better recommendation.

Technical introduction

I am shortly going to look at the MLF's submissions on the technical background, where he sought to give a particular, and we say entirely wrong, view of what an ANN is, and how he says it can be carved up between a general purpose structure and the weights and biases.

Before doing so, I think it is helpful to take a quick detour via the Act and the question of construction as to what a computer program is, so that we know what the targets are.

THE STATUTE

I'd like to start with by turning to **the statute itself [AUTH/28/p567 (bundle)].**

s1(1) says that a patent may be granted for an *invention* in respect of which certain conditions are satisfied.

The fourth of these is that the *invention* is not excluded by sub-sections 2 and 3 (also 4A).

And subsection lists out the prohibitions, with the 'as such' provision at the end.

s.125 says that the invention is, at least in general, that which is stipulated in the claim.

Now that is all obvious, but it makes a really important but very simple point – what is critical is to focus on the claimed invention.

And the importance of that is that one has to have regards to the invention *at the level of generality at which it is claimed*.

In this case, the invention is to a system or method of using a system containing *inter alia* an ANN.

It is common ground that an ANN is a structure of nodes, link, weights and biases.

In a hardware ANN, that is a box of electronics – Judgment [14].

And as the Judge held at [18] and [53], in a software ANN the same structure is emulated as if it were a hardware ANN on a software platform. And MLF confirmed that he agreed with this yesterday.

The point about discussing a hardware ANN, is that this is a convenient way of understanding what an ANN itself – the thing at the level at which it is claimed – is, free from any confusion which might be caused by the presence of the software platform underlying it.

As it is, MLF has rightly focussed his submissions on the ANN as a structure at that level, and quite rightly does not – at least as far as I can tell – seek try to make any point about the fact that a software platform can be used in some implementations.

So in answer to ML LJ Arnold's question yesterday, the point about a physical ANN is that it helps clarify about the nature of ANNs themselves, free of other distractions.

The other way this can be viewed is the important passage from **Vicom** at [16] – which are important for understanding here, and if I could just go to those now.

[Auth/13/p.328 (bundle)] [15] & [16] – which are the passages most widely cited in the authorities.

This is all important stuff! But there are **three** points I would particularly emphasise:

> **First** – [**read out end of para 15**] – reiterating a computer running a program which is directed to a technical process is not within the exclusion.
>
> That is because the invention is then to the technical process. Not that process may be implemented via a computer, but it is not a claim to a computer program 'as such'.
>
> **Second –** [**Read out first para of 16**] – It is no objection that the invention can be implemented as specialist hardware or on a conventional computer, and it is wrong to distinguish between embodiments carried out in hardware or in software since those do not change the essential nature of the invention.

What the Board is getting at here is that if the invention is one which is agnostic of the exact manner of its implementation, then generally it will not fall within the exclusion since it will properly be characterised as a claim to that invention, and not to the manner of its implementation – whether that be by a computer program or not.

It is similarly or accordingly no objection that the invention can be implemented as specialist hardware or on a conventional computer. That simply does not matter.

Mlady & ML will obviously see the relevance of that to the present case.

So those are the two point about hardware ANNs, and they are really facets of the same issue – which is that one has to look at the invention at the level of generality at which it is claimed, and the fact that it could be implemented using a computer is no objection if it would otherwise be claimable.

Sorry – that was a bit of a detour from the point we were on – which was the Act and the fact that that requires a close focus on that which is claimed.

The second preliminary point is then the construction of the Act.

Yesterday there was a discussion over what a computer was, and ML LY Birss raised things like analogue computers – which I have been business Googling overnight, but am still not much the wiser!

I can see that there could be an interesting debate about the wider edges of what a computer is, but what is critical to this case is what a 'program for a computer' is.

That is, or at least I think was, common ground.

The Judge addresses this **in [36]**.

He there sets out four definitions, each of which is broadly a sequence or set of instructions for a computer to cause it to do a particular thing.

And you will note that underneath, the UKIPO accepted these definitions at the hearing, and moreover have not sought to appeal them.

Indeed they put in some of the same in their skeleton at ϕ63(ii).

And as we point out in our skeleton at ϕ34, the description in *Gale* is possibly more accurate in that it refers to a series of instructions capable of being followed by a CPU; which is just a further clarification of what the word 'computer' in those dictionary definitions should itself be understood to mean.

And as we also note there in our skeleton, the language of instruction is also used in *Aerotel*.

That is why, in respond to a question of ML LJ Birss to MLF yesterday, there is a focus in this case on the 'instruction' – since it is agreed that that is what the statute is all about.

Further, as we will come to, MLF wider construction which he posited on his feet which ran 'input, mathematical processing, output' [T1/70$_9$] leads to what we submit are absurd results.

The point then is really what an instruction is, as to which we say that this is a term freighted with two elements:

> **First,** that there is an element of command or order about it. It is something which tells someone (or here something) else what to do.
> **Second,** in particular in this context, that the instruction is a human direction as to what to do.

Tech background proper

With those introductory comments, I can now turn to the technical background proper.

As we set out in our skeleton at §11, the Judge addresses the technical background in [14] – [18], ... and **Prof Pardoe** sets out the basic structure of an ANN in his paragraphs 19 – 43 [SUPP/tab 9/ p.212].

Yesterday MLF sought to give YL's a very carefully curated view of an ANN, designed to support his submission that an ANN can be divided into a 'general purpose' structure of nodes and links, with a 'program' of weights and biases laid on top of that.

MLF seeks in his submissions to split the links and nodes, from the weights and biases, to say that the former is a computer and the latter the program.

He was at great pains to give the impression that there was a generic or general purpose set of links and nodes, which was programmed with weights and biases.

But we submit that that is artificial and wrong, for six reasons.

CHANGING STRUCTURE IN TRAINING

First, I would like to start with an important point, which MLF did not bring to your attention, which is that during training the **structure** of the **links and nodes themselves** may be changed.

If I could show you that – go to Pardoe ϕ39 [SUPP/9/p.216]

It is a short piece of evidence, but given the new way that the UKIPO now puts its case, it takes on some real importance.

What this underlines is that it is not so simple as MLF seeks to portray it – in training the nodes *themselves* (and therefore *links too*) may be pruned.

So, for example, in training some ANNs may prune nodes and therefore alter the remaining weights and biases in one way, and another ANN may prune different nodes, or none at all, and therefore alter the remaining weights and biases in another way.

So the question of whether and what nodes to prune (if at all, sure) and what weights and biases to ascribed are all interlinked.

MLF is just wrong to suggest that in ANNs the links and nodes are a fixed given, and the weights and biases the only variables.

Whilst Prof Pardoe says that this may only occur in **some** cases, the importance of the point is what it tells us about the nature of ANNs themselves – namely that it is not right that the training process is limited to only weights and biases, but rather the structure of the ANN is a composite whole. The point is that that the elements of that structure are interlinked and cannot be divided up as MLF would like to have it.

Weights and nodes are inherent features and cannot be divided out.

Second, but relatedly, it is simply not possible to have an ANN which does not have all of the features – links, nodes, weights and biases.

You cannot remove the weights and biases from an ANN at all. Every link must have a value and likewise every node, since it is a feature inherent in being a link or a node.

It is quite unlike a computer, where one can download or remove programs at will, and the computer remains.

Again the point is that these features are intertwined.

Function is defined by its structure as a whole.

Third, relatedly, the function of an ANN is determined by its structure as a whole.

The links and nodes only work with the *specific* weights and biases, and *vice versa*, and they are all entirely intertwined.

You cannot take the weights and biases from one ANN and run them on another ANN – it just doesn't work, or even really make any sense as a discussion.

But that is because an ANN can only function as a complete machine.

And where MLF yesterday went on about his examples of training one ANN and rolling it out across other ANNs, what is important to understand is that in each case you are rolling out the same complete package of links, nodes, weights and biases. You are not rolling our weights and biases on arbitrary generic ANNs, but simply replacing the entire complete thing in all its elements.

There is no evidence that anyone else anyway decomposes an ANN in the manner the UKIPO now tries to do.

Fourth, there is no evidence that anyone, other than the UKIPO in this court, has ever sought to characterise or decompose an ANN in the way that the UKIPO does.

In his report Prof Pardoe describes it as a machine as a whole, indeed saying that after each iteration during training the **new machine is a better machine**. That is his report ϕ24. He does not describe it in terms of an improved program for a machine.

And nor does he anywhere suggest that an ANN can be described as a general purpose computing part with a program overlayed.

Indeed you may be aware that the OECD definition of an AI describes it as a 'machine-based system', not a computer or computer program, and that same definition has been taken into the new EU AI Act too.

Further the UKIPO is unable to cite anyone one else who describes an ANN in the manner they do either.

Given that they are advancing an argument which is completely new, and indeed a reversal of their original concession, one would have expected them to come up with articles or the like explaining the operation of an ANN in the terms they now urge on the Court, but they have nothing (and it is not like they are scared of a hand-up or two, or indeed referring to trade articles as they have included in the bundles already). But on this critical point, they have nothing.

And we do pray in aid, not only that lack of any support for this new point (one we note not supported by Prof. Pardoe), but that this was not a view which was espoused by the hearing officer, or by the UKIPO in front of Mann J, where the UKIPO explicitly took in fact the exact opposite position – a position which was no doubt entered into with careful consideration and upon advice.

All of that indicates the artificiality of the UKIPO's position, which has no grounding in any evidence, and is not supported by the only evidence in this case from Prof Pardoe. *Pardoe ϕ24*

FUNCTION vs COMPUTER PROGRAM

Fifth, the argument of the UKIPO risks confusing the function of a device and whether or not there is a computer program in that device.

The point is this – the fact that a device is able to do something which could be described mathematically and outputting a result, does not mean that it is a computer or more relevantly using a computer program to do so.

MLF put it as 'it inputs, mathematical processing, outputs' – that was at [T1/70$_9$].
But we submit that that is a flawed and unreliable guide.

The basic point is this. You could create an electronic device the function of which was to perform a particular process, which process could be described mathematically and to output an answer; and you could write a computer program to control a computer to perform the same process, even to do so in the same way.

But that does not make the electronic device a program for a computer – it is just that different approaches are being used to achieve the same function.

Thus one has to be very careful with equating the function of a device – even a function which can be described in mathematical terms – with whether there is a computer program in it.

So to take a simple example, a basic ohm-meter (that is a device which measures the electrical resistance of things) uses a collection of simple electronic components to take in an input, do an analysis of it which if you were so minded could be described in

purely mathematical terms, and gives an output being the position of the needle on the dial.

The function it performs is a result of the particular values ascribed to the network of electronic components within it.

To use MLF's submissions from yesterday [T1/70$_9$].: it inputs, mathematical processing, outputs.

On MLF's approach, a basic electronic ohm-meter would be a computer with a computer program within the Act, but in our submission it is without doubt that that is **not** the kind of thing which the legislature had in mind.

Sure the structure of an ANN is much more complex, and so therefore also its analytical capabilities, but there is no qualitative difference between that and the ohm-meter.

Furthermore an electronic device of a little more complexity, say some kind of **hardware filter arrangement for an electronic circuit**, may well be designed by an iterative process by which the parameters of the resistors, capacitors and amplifiers involved are modelled on a computer and tweaked to optimise the filter spectrum, and then put into production afterwards.

Just to unpick that, in an electronic device such as a mobile phone the transmission is received at a certain frequency, but the received message may be interfered with by a plethora of other signals at other frequencies. What you can do is use arrays of capacitors, resisters, things called inductors etc which have the effect of draining off or blocking the unwanted frequencies, but allowing the one you are after through.

These are all electronic components arranged normally in one of a number of well known structures to achieve this function, but the fine-tuning in any case involves a decision of the exact values to ascribe to each individual component.

On MLF's case such hardware filters would be considered computers with computer programs (being the values assigned to the components), and yet in my submission that would be considered an extremely surprising conclusion, these are just standard electronic devices and plainly not what the Act was ever contemplating as having or being a computer or a computer program.

One could even go further – a **sextant** for example?

That is a device, there purely mechanical, which takes in an input, performs a translation which can be described if one wishes in purely mathematical terms, and spits out an output.

A sextant has a basic structure, but it is the markings on the various bars and levers which enable it to perform its function.

Again, as he puts it, input, mathematical processing, output.

Sure, that is not even electronic, but is it not, on his case, a computer and a computer program? There I think the computer program would be the markings on the bars and arms of the sextant.

The point we are making here is not really that these are computers and computer programs, but that MLF's approach of asking what *function* the device has is an unsafe way of identifying a computer and, more particularly, a computer program.

And further it shows that MLF's suggestion that what mattered under the act was 'input, mathematical processing, output' [T1/70$_9$] leads to what we submit are absurd and unprincipled results.

The components are not sending instructions, and to what?

Sixth, another point about MLF's case is to dig a bit deeper as to what he really means when he says that the weights and biases of an ANN provide the computer program.

Let us take a hardware ANN, which can be implemented in pretty simple forms.

Links can be just tracks of wire or the like.

The weights can just be an amplifier or resistor on that track.

The node itself can be implemented as a simple amplifier, with the bias an addition similar component.

So on MLF's case, some of the amplifiers and components are the computer and other the program.

But they are all interlinked and intermingled in one complex structure.

Furthermore one can ask what are the 'instructions' on his case, and what are they 'instructing'?

What he refers to are the individual current flows through the resistors and amplifiers.

But they are not passing instructions to anything at all. All they do is their normal electronic job of reducing or amplifying the current passing through them; they just function to restrict or increase the current *passing through themselves*. Weights and biases tangible "things" which affect structure and function, and not a program for a computer as MLF would have you believe.

Sure this may have a downstream effect on another component, but a downstream effect is not to be confused with an instruction to that component to do or not do something. It is not telling it do something, but rather it is just performing its function with a consequential downstream effect, as happens in electronics. Instructions and consequences are functionally different concepts.

Intractable problem

The **sixth** point to pick up here again is that the operation of the ANN is fundamentally different from a computer program, not only in the lack of any series of instructions, but in that such function as is present is not in the form of a series of logical instructions defined by a programmer. That is the intractable problem point which I have already addressed you upon – the human in most cases (of which this is one) has no idea how to address the problem, let alone to write a series of instructions to solve it. To put this in the context of the invention and musical texture or timbre, there is no definition of what amounts to "timbre" – it is "ill-defined" in the language of the ANN arts - so the programmer cannot program instructions that define this characteristic. However, within the ANN of the invention and in practical terms, the invention can approximate musical texture similarity ("timbre") without needing a strict definition of a timbe which, in any event, doesn't exist.

As I have submitted the term 'instruction' is one freighted in this context as human instructions, and here in an ANN there is nothing like this.

At most the human is involved in setting the parameters for the training program – and we do accept that there is a program involved at that stage. But once you turn to consider the final ANN itself, the human is not, and in any case other than the entirely trivial never could instruct the trained ANN what to do or how even to attempt to solve the problem at hand.

The Judge made a similar or related point in [77] and [78] of his Judgment, where he said that even if the UKIPO was right that there was a computer program at the training stage, then in any event the result of that training process is itself a separate technical thing – the trained ANN.

© 2024 Hepworth Browne