User-Centric Computing for Human-Computer Interaction

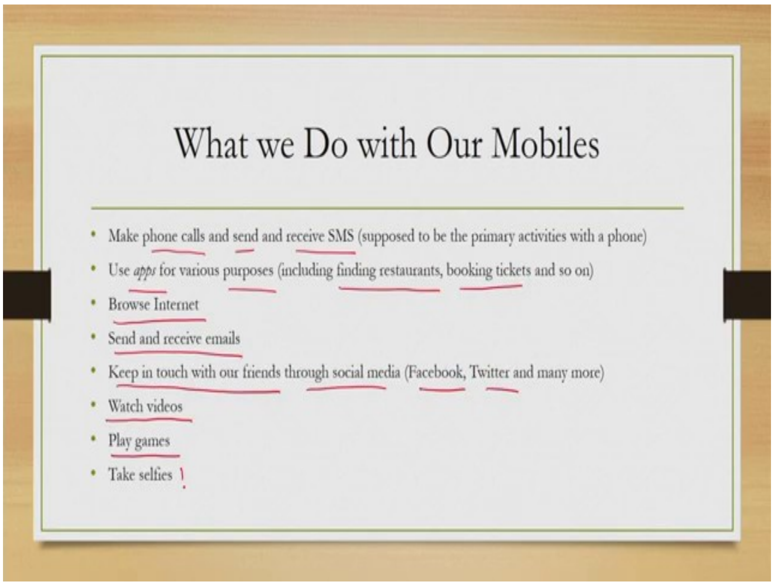
Prof. Samit Bhattacharya
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati

Lecture - 21
Mobile typing models (single finger and two thumb typing)

Hello and welcome to lecture number 21 in the course User Centric Computing for Human Computing Interaction. So, first let us have a recap of what we have learnt so far. So, if you may recollect we are now discussing computational user models that are applicable to contemporary systems and interfaces that we see all around us. So, in the previous few lectures we covered 2 D and 3 D pointing models or more technically which are known as bivariate, and trivariate pointing models. Then we covered models for constraint navigation there we discussed 2 models one is the Steering law and the other one is menu selection model.

Today we are going to discuss few more models related to the contemporary interactive systems and these models are applicable to a very popular interactive activity that we regularly perform. So, these are models of mobile typing which anyway as all of us know is a very frequent activity that we perform with our phones.

(Refer Slide Time: 01:53)



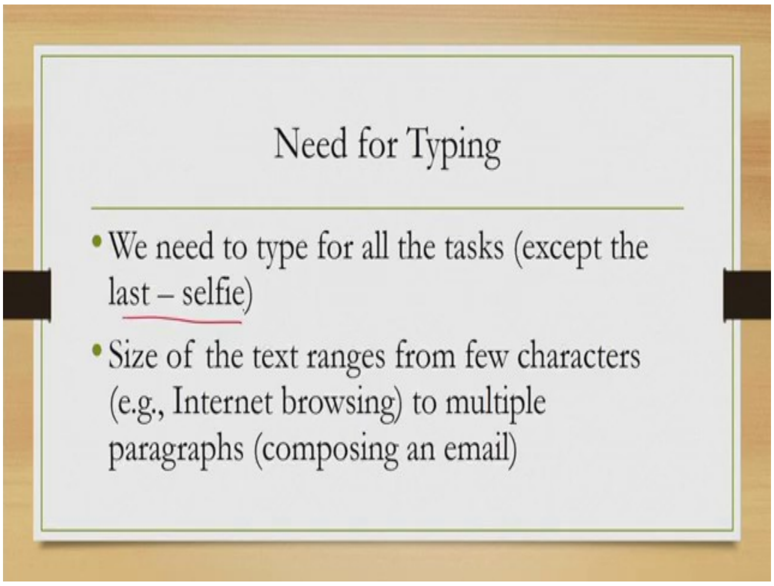
So, when we are talking of mobile typing first let us try to understand what we do with our mobile phones. So, if you may note down what are the things that you do regularly with your phone, you may find out something like this list. You may figure out this list of activities that you regularly perform with your mobile phone. The most obvious activity is of course, making phone calls and sending and receiving SMSs.

Now, these are of course, the primary activities that are supposed to be done with your phone. However, as all of us know these primary activities are nowadays getting less and less attention and we primarily use our phone for many other activities that are not related to this primary tasks; namely we use apps for various purposes including finding restaurants, booking tickets and so on.

We browse internet, we send and receive emails, we keep in touch with our friends through social media such as Facebook, Twitter which is probably the most popular activity nowadays particularly with the younger generation. We also watch movies or videos in our phone, we play games and of course, another very frequently performed activity is taking selfies which nowadays as become like an addiction.

So, these are some of the major activities that we perform with our phone.

(Refer Slide Time: 03:29)



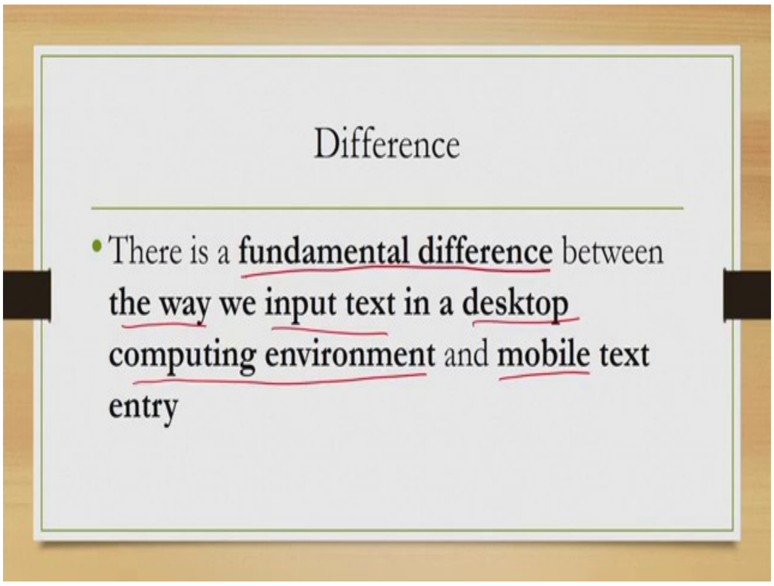
Now, among them except the last one that is taking selfie, if you note all the other activities require us to type something to enter some text. Now the size of the text varies. It may require us to some activities may require us to enter few characters such as entering URL in a browser or it may require us to enter large amount of text such as composing and sending an email.

(Refer Slide Time: 03:54)

So, text input is a very important part in our interaction with mobile phones and that is true for any mobile devices that we are interacting with nowadays mobile phones or tabs.

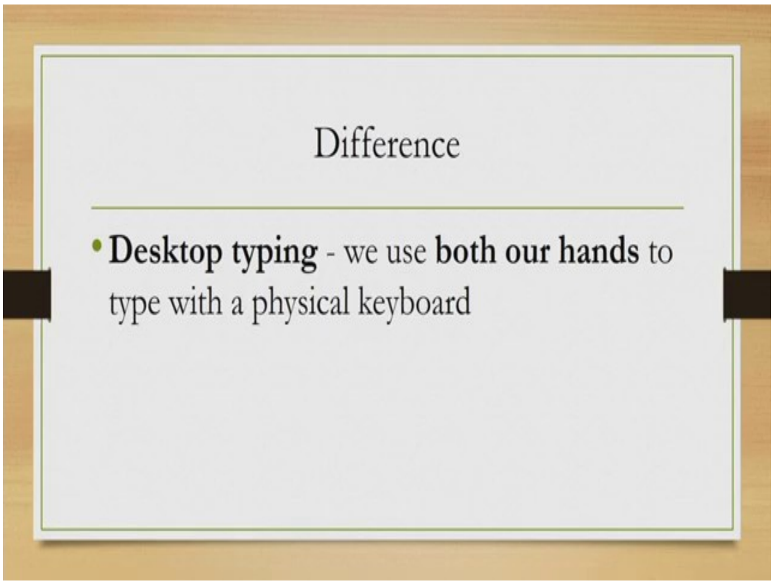
So, it is pertinent to model the typing performance in the context of typing on a mobile device so that we can utilize the models for better design of such typing interfaces. Now before we go into that discussion on mobile typing models we may need to understand or we would like to understand the difference between typing with a desktop computer and typing with a mobile phone, are there any difference or we behave in the same way in both the cases.

(Refer Slide Time: 04:45)



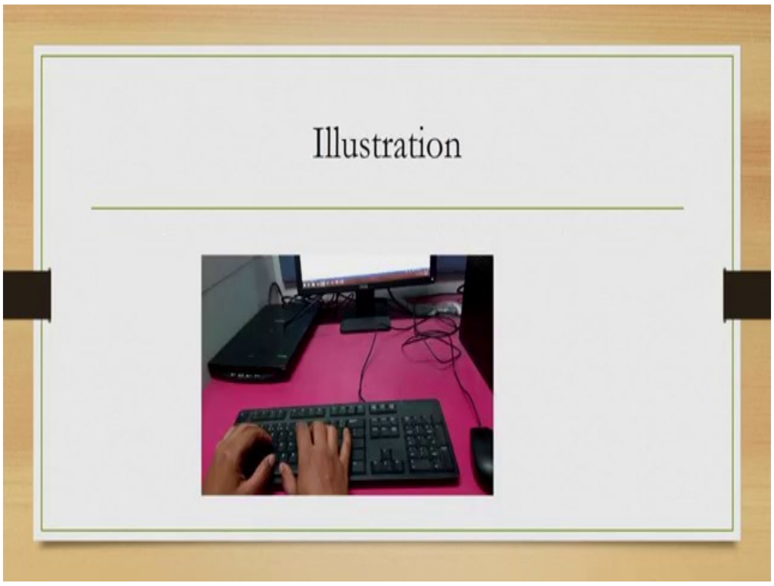
Now, there is actually a fundamental difference, difference in terms of the way we behave when we input text in a desktop computer or in a desktop computing environment and when we input text in our mobile phones or when we perform a mobile text entry task. So, this fundamental difference is in terms of how we use our hands, how we use our hands or fingers to input the text.

(Refer Slide Time: 05:11)



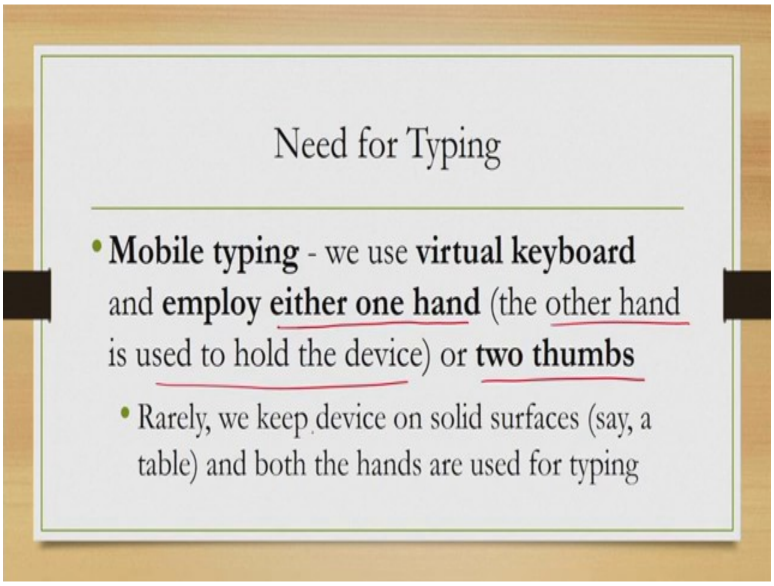
In the context of desktop typing typically what we do is we use both our hands. So, the physical keyboard typically rests on a surface typically your table or desktop. And we use both the hands on this keyboard to enter the text which looks something like this.

(Refer Slide Time: 05:31)



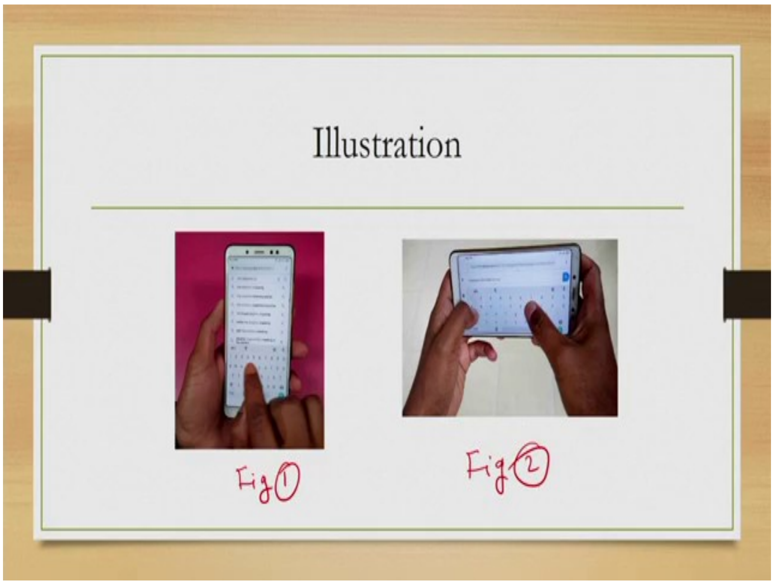
In this figure as you can see we are illustrating the typing in a desktop environment, here both the hands are on the keyboard and both all the 10 fingers of both the hands are utilized to type the text. Now what happens when we type with a mobile phone?

(Refer Slide Time: 05:50)



The way we use our hand or finger is different if you may have noted. Here what do we do is typically use either one hand whereas, the other hand is used to hold the device or we use 2 thumbs not the 10 fingers to type. It is very rare that we keep the device on solid surface and use both our hands almost nobody does that.

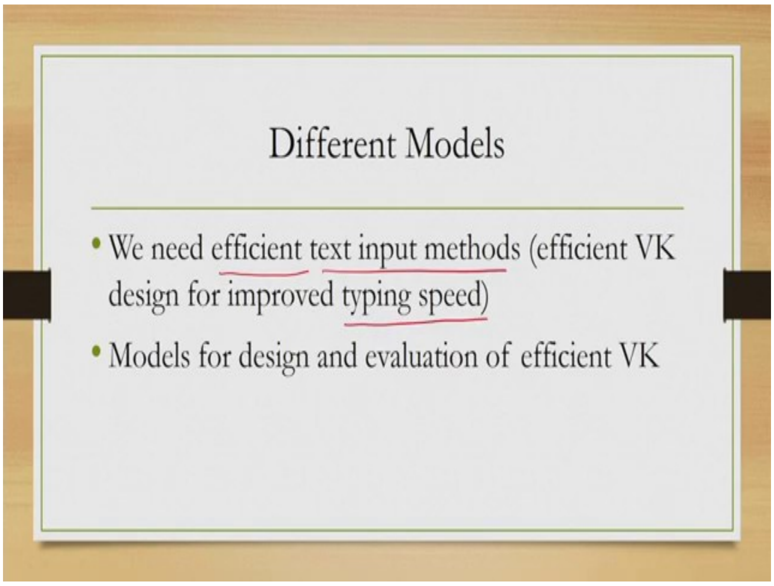
(Refer Slide Time: 06:18)



So, the situation is depicted in these 2 figures. If you note in the first figure, in figure 1 it is shows a single hand typing. So, one hand is used to hold the phone and the other hand or the finger of the other hand is used to type; so, single finger typing or single hand typing. In figure 2, in the other figure we see two thumb typing; so, both the hands are used to hold the phone as well as type where the typing is done with 2 thumbs rather than any other finger.

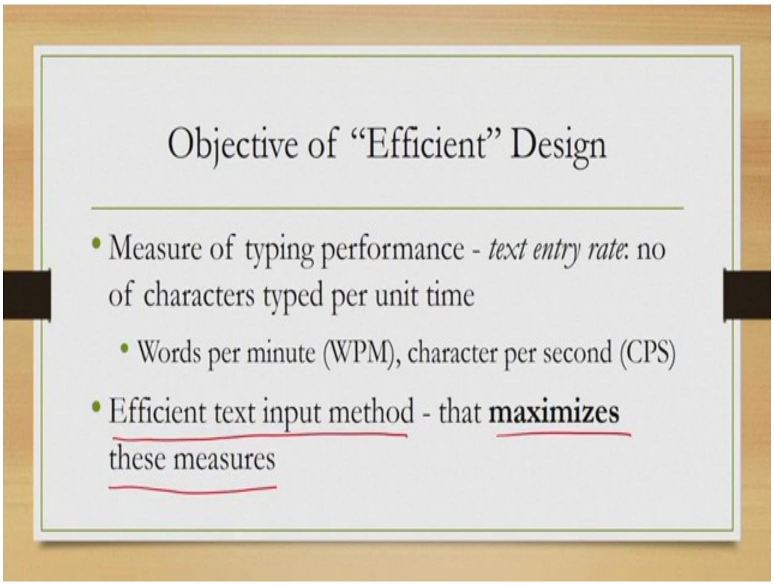
Now these are most common typing behavior that we observe in mobile text entry. So, we need some efficient text input methods that actually utilize this typing behavior that actually assists in this typing behavior.

(Refer Slide Time: 71:22)



So, when we say efficient, so, this efficiency is in terms of typing speed. So, how fast we can enter the text. So, our objective is to have an efficient text input method or efficient way of performing mobile text entry that takes into consideration the different typing behavior and provides us a way to improve the typing speed and this method can be obtained with the help of models for typing performance in mobile environment. So, we will see how and we are going to discuss couple of such models in this lecture.

(Refer Slide Time: 08:01)

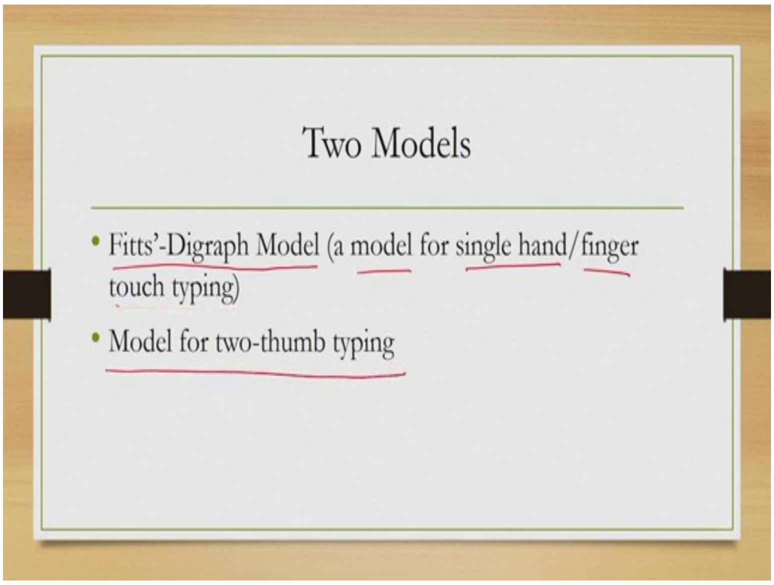


So, when you say typing performance as I said typing rate is the measure of the performance.

Now, this measure can be represented in many ways, the most common representation is words per minute or characters per second, typically will stick to these to measures. And when you talk of efficient text input method what we mean is essentially that a method that maximizes these measures; word per minute or character per second.

So, when we have a text input method that maximizes the text entry rate represented in terms of word per minute or characters per second we refer to it as efficient text input method. So, when we are talking of modeling these typing behavior there are 2 models that are reported in the literature there are actually many models, we will have a discussion on 2 of these models.

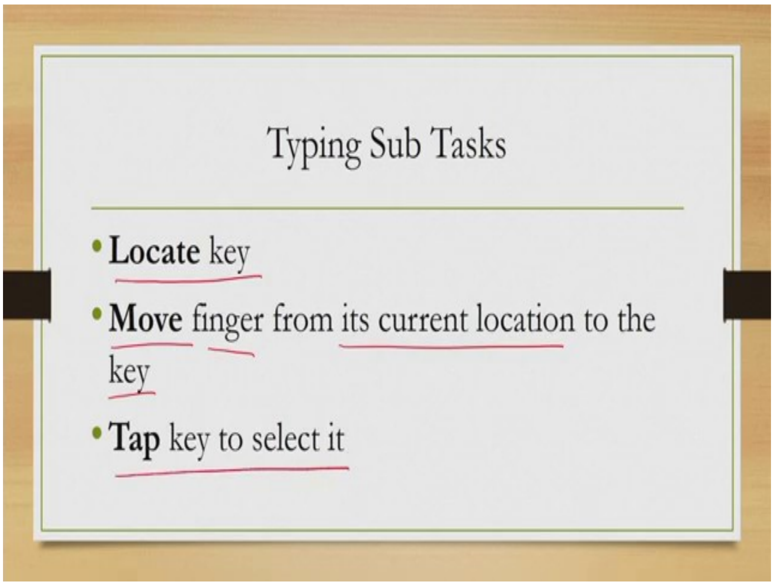
(Refer Slide Time: 08:58)



One is the Fitts' - Digraph model which is a model for single hand or finger touch typing which we have illustrated earlier. So, the Fitts' - Diagraph model can be used to model the performance of single finger or single hand typing on a mobile device and we have a separate model for two thumb typing so, both these models we are going to discuss in today's lecture.

So, let us start with the first one that is the Fitts' -Digraph model.

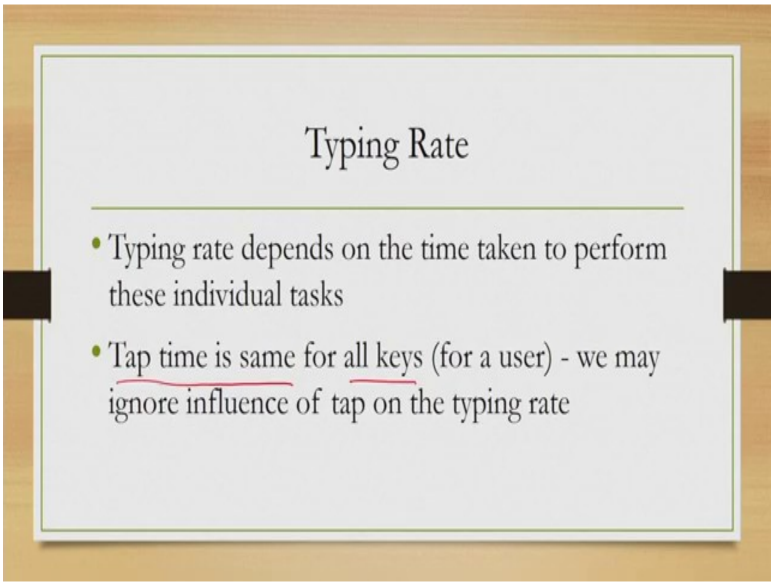
(Refer Slide Time: 09:29)



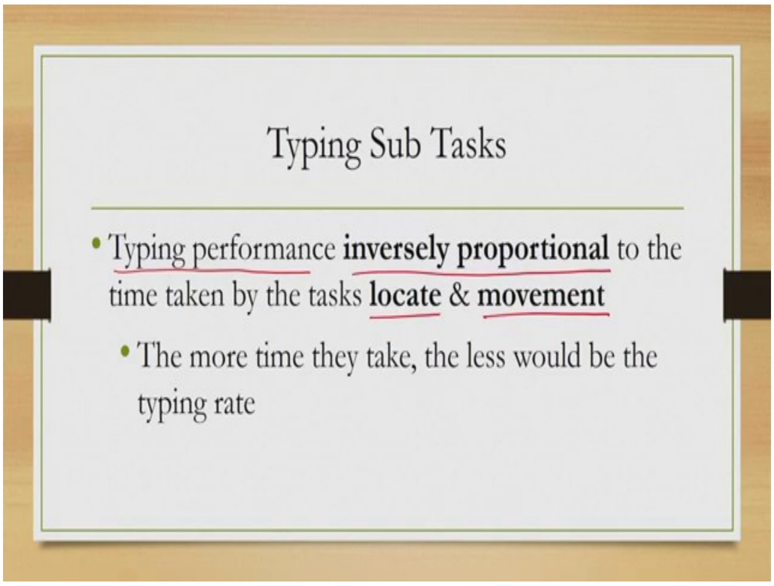
So, when we say that we are going to type a text what are the sub tasks that are implied. So, typing a text involves 3 sub tasks. So, here we are assuming that there is a virtual keyboard through which we will be typing. So, when the virtual keyboard is there and we intend to type a text, so, it implies 3 sub tasks. One is we have to first locate the key containing the particular characters that we want to type, second is we have to move our finger from it is current location to the location of the particular key which we want to type and third one is the selection of the key through tap.

So, we are given a virtual keyboard with this keyboard we want to type a text and in order to type the text we need to perform 3 sub tasks. The first sub task is to locate the key; that means, among all the keys we need to visually locate where the key is, then in the second sub task we move our finger from it is current position to the position of the key and in the third sub task which is the final task we tap on the key to select the character.

(Refer Slide Time: 10:48)



Now, typing rate depends on as is obvious the time that is required to perform each of these tasks. So, the total time required to perform all this sub tasks determines the overall typing rate; however, if you may note the tap time may be considered to be the same for all the keys for a particular user. So, it not a variable component and we can ignore it is effect to come up with a model for overall typing performance. So, essentially what we are going to do is model the performance in terms of the other 2 task performance namely locate the key and move finger to the key. (Refer Slide Time: 11:34)

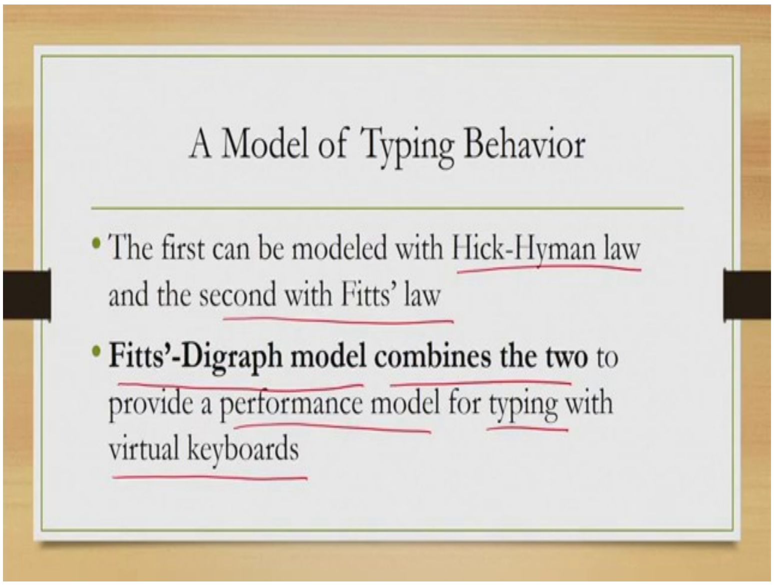


What we can say in fact, that is an intuitive knowledge also that the typing performance is inversely proportional to the time taken by the tasks locate and movement. So, if I take more time to locate the key then definitely it is going to affect my typing performance. So, my typing performance will reduce because I am taking more time to locate the key. So, my overall text entry rate will reduce.

Similarly, if I am going to take a more time to move my finger that also is going to reduce my typing performance in terms of text entry rate as is very obvious. So, essentially both of these quantities inversely affect the performance. So, the more these values are the less is the performance and the less these values are the more is the performance.

Now what is our objective? Our objective is to come up with a model of typing performance, in other words our objective is to model the performance in a way such that we can compute the typing rate with this model. So, in order to achieve this objective we need to individually model the 2 sub tasks namely locate and move. So, if we can model the sub tasks, then we can combine these models to come up with the overall model of typing performance.

(Refer Slide Time: 12:46)



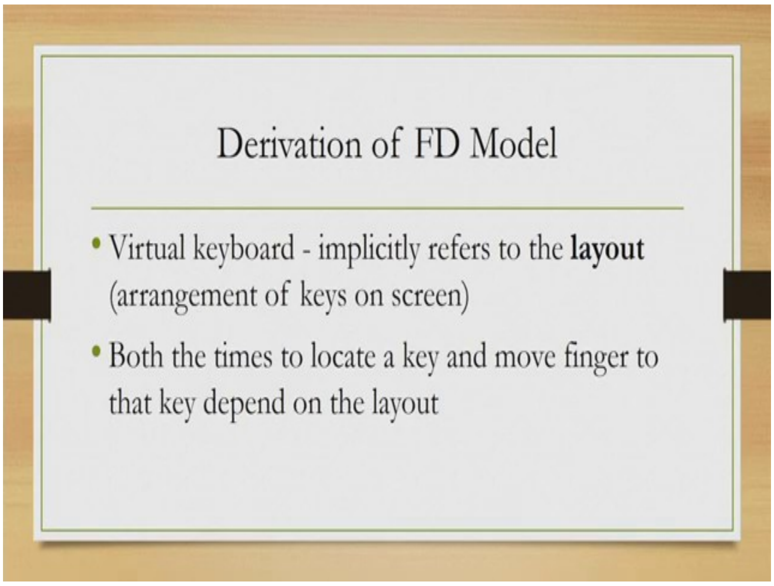
Now, as you can see in case of locating a key in the virtual keyboard what we need to do. So, essentially there are keys so, the virtual keyboard contains a set of keys among these keys we have to figure out where the particular key is. So, essentially we are trying to react to the stimulus given to us that is the layout and our task is to locate or decide where the key is that is our reaction time. So, this entire activity of locating a key in a set of keys can be modeled with the Hick - Hyman law which we have discussed earlier.

The other task is moving the finger from the current location to the target key location, here as you can see this is a motor movement or manual movement task it refers to movement of finger which is a motor organ from one location to another which can be modeled using the Fitts' law in fact, which can be modeled with the classical Fitts' law.

So, this overall Fitts' Diagraph model is a combination of the 2 individual models. So, Fitts' Diagraph model can be considered to be a combination of the 2 individual models, one is the model for locating a key that we can model with the Hick Hyman law. Other is the model for moving the finger from current location to the target key location which we are modeling with which we can model with the Fitts' law and in the Fitts' diagraph model we combine these 2 models to come up with a performance model for overall typing performance in the context of virtual keyboards on a mobile device.

So, let us now try to go into the details of the model, let us now try to understand the components of this model.

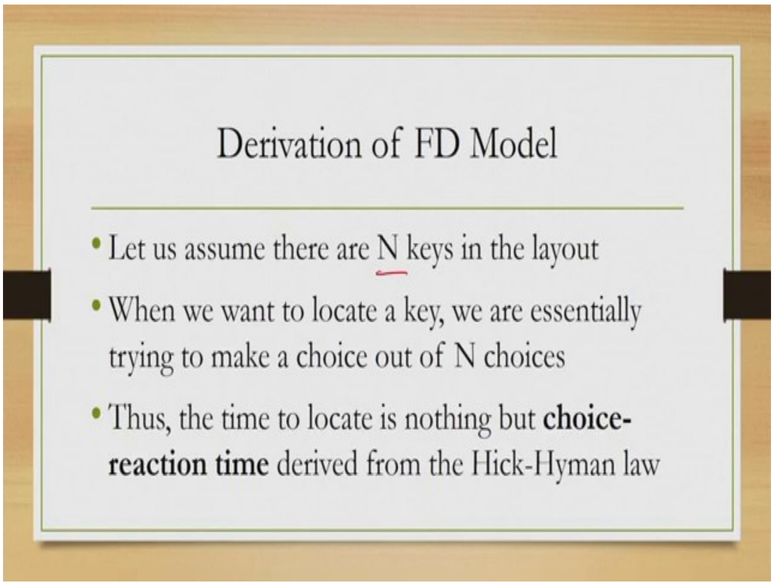
(Refer Slide Time: 14:48)



So, when we are talking of the model we are talking of modeling typing performance for a given virtual keyboard layout which should be displayed on the screen of a mobile phone or mobile device which may be a phone may be a tab. And here we are assuming that the typing is performed in a particular way, we are holding the device in one hand and using a single finger to type. So, this is the particular typing behavior we are assuming to model the typing speed and the model is applicable only for these behavior. So, if you are typing with any other behavior and trying to predict the typing rate then this model will not work.

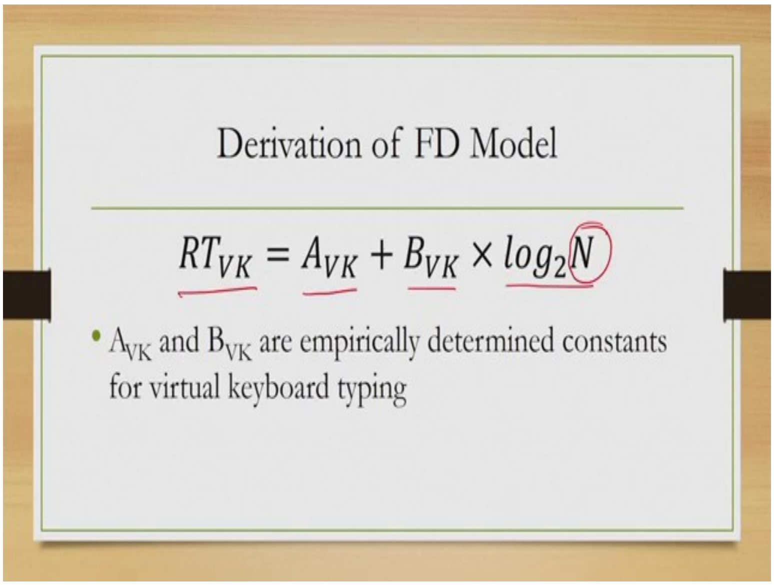
Now, when we say that we are having a virtual keyboard layout what we refer to is essentially that the typing speed which is dependent on the time to locate the key and time to move the finger is dependent on the positioning of the keys on the layout.

(Refer Slide Time: 15:47)



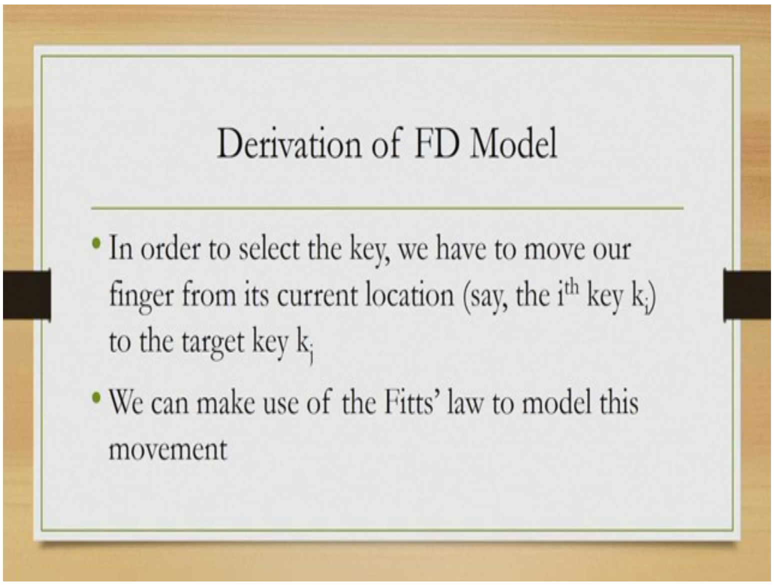
Now, let us assume that there are N keys on the layout. So, when we want to locate a key we are essentially trying to locate one key among these N keys. In other words there are N choices which are given to us and out of these N choices, we are asked to make one choice that is locate one key. So, essentially it is the same as a choice reaction time task, here our objective is to determine the reaction time to model the reaction time and this reaction time we can model with the Hick - Hyman law and we will also assume that the probability to select a key is same for all the keys.

(Refer Slide Time: 16:36)



So, if we make that assumption then we can represent the choice reaction time or the time to locate a key with this formulation based on the Hick - Hyman law that is RT VK that is the time to locate a key on the virtual keyboard layout as $A+B*\log_2 N$, where N is the number of keys on the layout, A and B are constants. In the lecture on Hick - Hyman law we have discussed how to determine the values of these constants through empirical studies and we can follow the same technique to determine these values in the context of mobile typing.

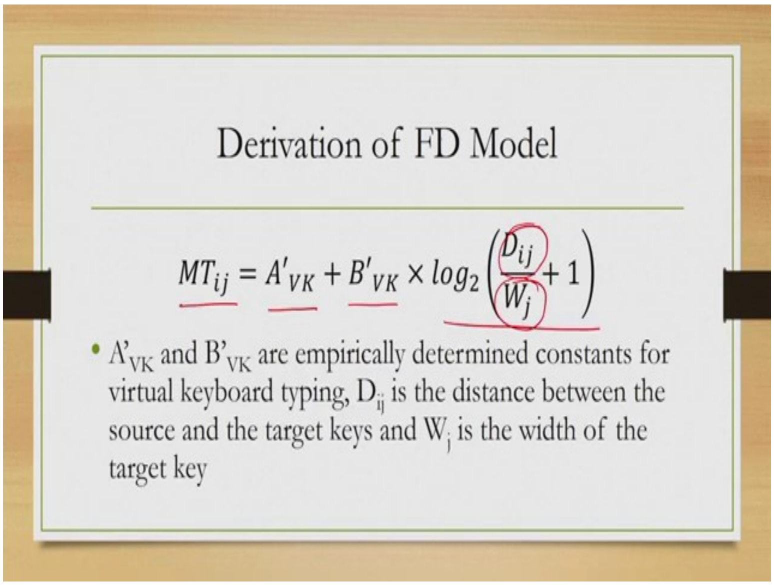
(Refer Slide Time: 17:15)



The next task is to select the key. So, once we have located we need to select it, so, for that we need to move our finger from it is current position to the position where the key is and that as we said we will model with the Fitts' law. So, let us try to come up with the generalized model.

So, suppose we just selected the i th key represented by k i this notation and we want to select the next j th key represented with the notation k j. Now the time required to select the key time required to move the finger from the last key selection location that is k i to the target location that is k j can be represented with the Fitts' law in the following way.

(Refer Slide Time: 17:56)

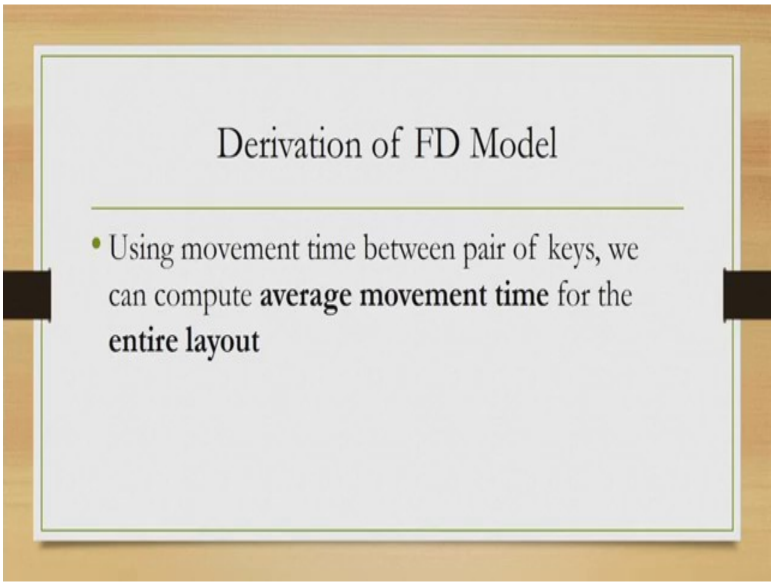


The movement time can be modeled in this way where MT ij is the time to move the finger from the i th key to the j th key which is a Fitts' law model; A dashed and V dashed are constants and this is the index of difficulty for the movement as we discussed during the discussion on Fitts' law, D ij is the distance between the 2 keys and W j is the width of the target key.

Similar to the Hick - Hyman law in the Fitts' law modeling also there are 2 constants A dashed VK and B dashed VK and these 2 constants we need to derive through empirical means. Now these empirical means we have already touched upon in our earlier lectures on Fitts' law. So, we will not be going into the description of these studies again and if you are interested you are advised to refer to the previous lectures.

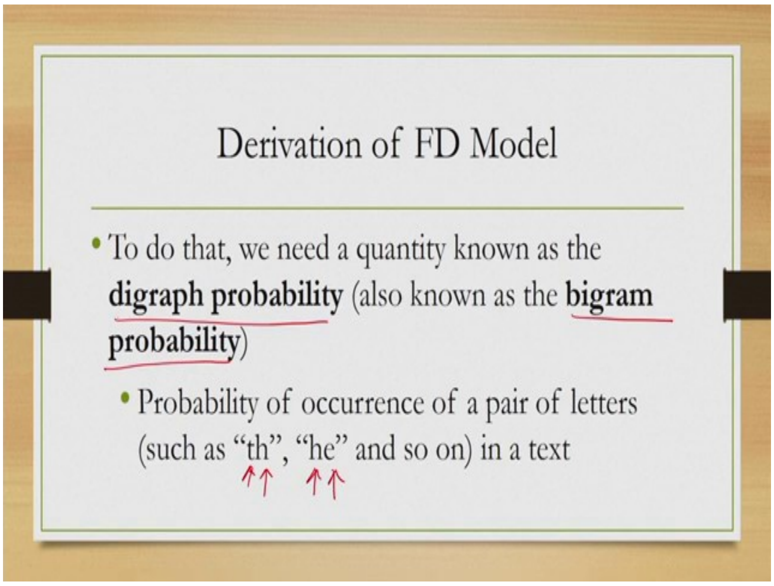
Once we manage to model the movement time between any arbitrary pair of key we can use this model to actually come up with a model for average movement time over the entire layout that means, average movement time between any pair of keys.

(Refer Slide Time: 19:04)



So, earlier model is a specific movement time model for a pair of keys and average movement time is applicable for the entire layout it will be the movement time to move from anyone key to any other key in the layout.

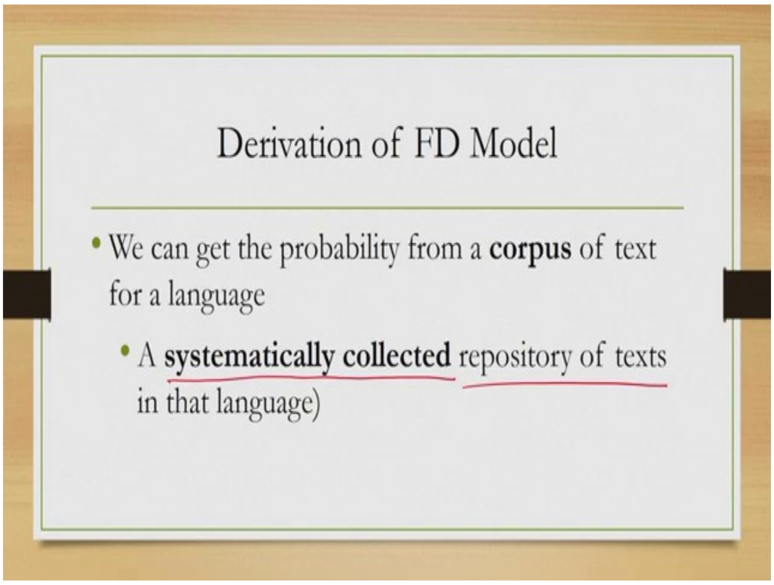
(Refer Slide Time: 19:17)



And that is possible by considering one more concept that is known as model of a language and it is technically called digraph probability or more popularly bigram probability.

So, it is the probability of occurrence of a pair of letters in a text. For example, in a English text the probability of occurrence of the letter t and h or h and e or any other such pair will be the digraph or bigram probability for these pairs.

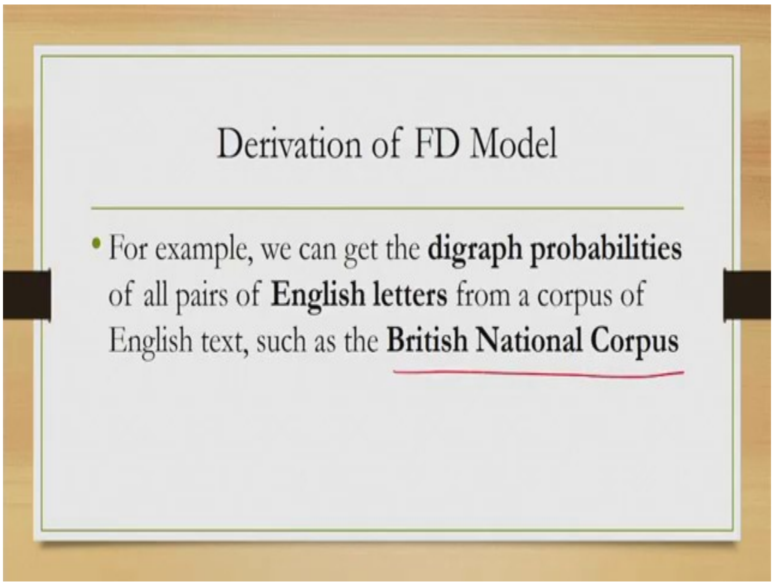
(Refer Slide Time: 19:58)



So, the objective is to come up with a average movement time model using digraph probability and the movement model based on Fitts' law between any 2 keys. Now the question is how we can find out this digraph probability, for that what we need is a corpus of text. The corpus is essentially repository of text which has been collected by following a systematic method and for particular language we consider a corpus to obtain the bigram or digraph probabilities of the character pairs that are there in the alphabet of that language.

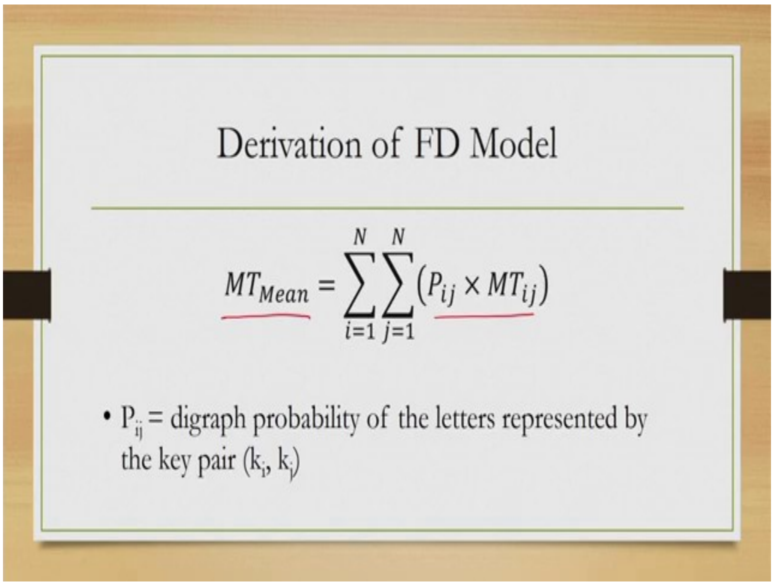
For example, suppose we want to know the digraph probabilities of all the letter or letters and numbers and other punctuation symbols whatever is there in the English alphabet if you want to know the digraph probabilities of all these pairs, between letters, between letter and number, between numbers, between letter and punctuation marks so on. Then we need a corpus of English language text and there are many such corpus available one very widely used corpus of English language text is the British national corpus or BNC.

(Refer Slide Time: 21:19)



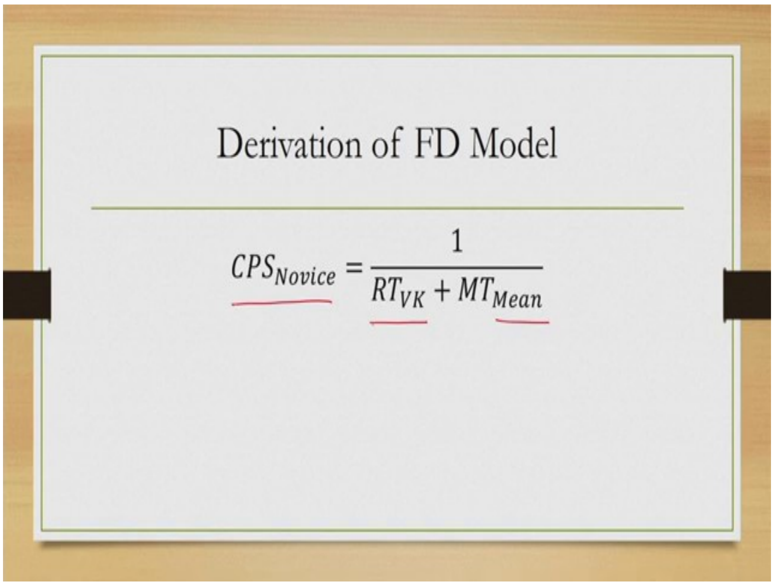
So, using this corpus we can actually find out the diagraph probability distribution of character pairs in English language.

(Refer Slide Time: 21:27)



So, now once we have this diagraph probability values, once we are having these probability values we can use it to come up with a formulation for average movement time between any pair of keys in a virtual keyboard on a mobile screen. So, that formulation is given here where MT Mean refers to the mean or average movement time for a given layout which is a summation of this quantity, P ij is the digraph probability for the key pair for the i th and j th key pair MT ij is the movement time or the time to move the finger from the i th key to the j th key and we sum this quantity for all the keys present in the layout. So, this expression tells us the average movement time.

(Refer Slide Time: 22:22)



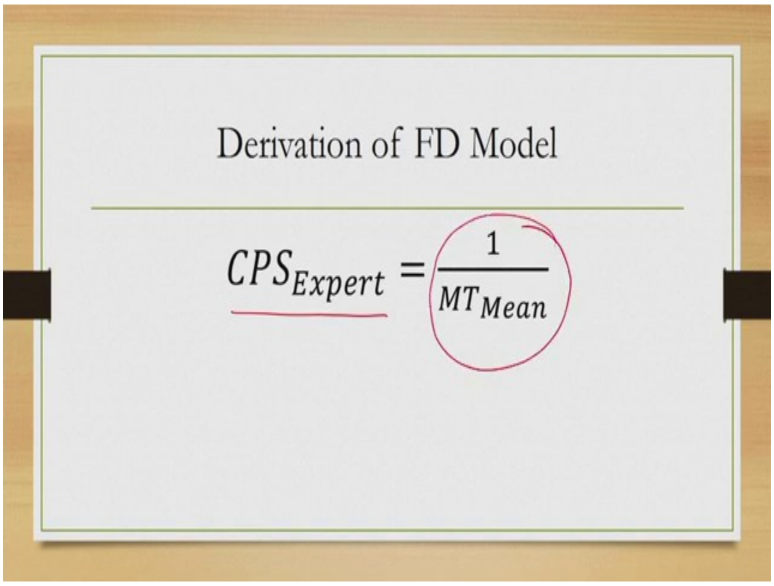
Now, with this average movement time notion and the choice reaction time which is modeled with the Hick - Hyman law we can come up with a model of typing performance. If we want to model the performance in terms of characters per second or CPS then we can get a model in this way CPS novice is $1/(RT+MT_{Mean})$.

Now, here novice means this model represents the performance of a novice typist or a typing on the layout for the first time. So, it is assumed that a novice typist will require some time to first locate the key so you have added this value and then move the finger. So, you have added this value; however, that is not the case when we are thinking of an expert typist.

In case of an expert typist it is already assumed because we are assuming the typist to be expert it is already assumed that the time required to locate a key is not there the typist already have learnt about the position and memorized it and that is called familiarity. So, because of this familiarity with the layout the typist can reach to the key without even looking at the layout and so the time to locate the key is negligible or non-existent.

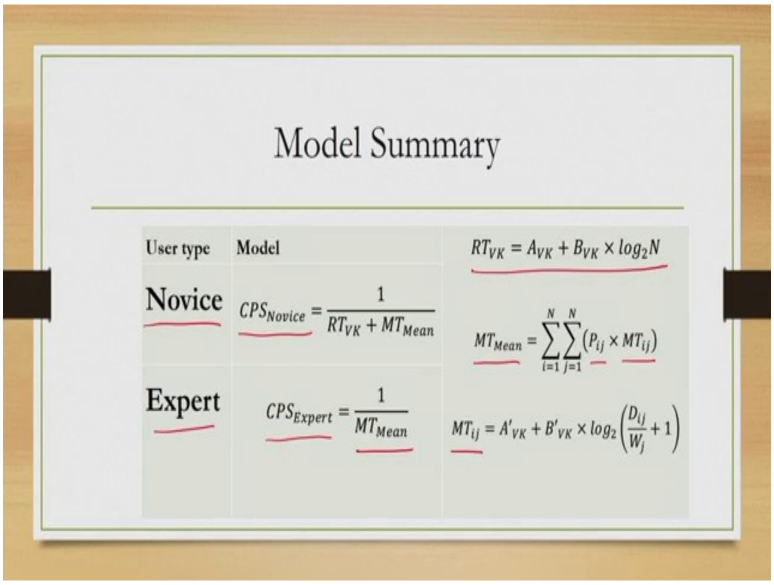
So, what we need is only the movement time model. So, in that case therefore the overall typing performance is represented with a separate expression.

(Refer Slide Time: 23:52)



That is for CPS of an expert user or a person who types regularly with the virtual keyboard, we can represent the performance model by simply $1/MT_{Mean}$. So, here the choice reaction time component is not there.

(Refer Slide Time: 24:11)

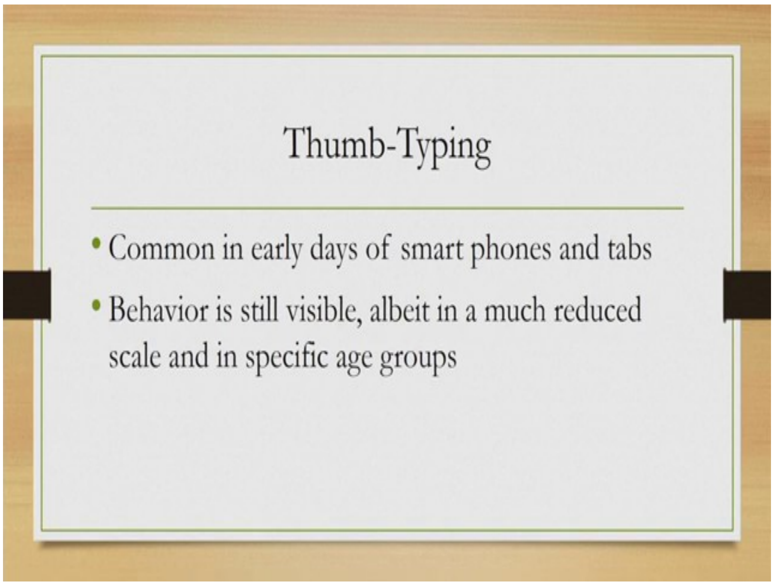


So, in summary what we have learned is that. So, there are 2 types of users we are assuming novice and expert, for novice user the performance can be modeled in terms of

CPS as 1 by RT plus MT mean for expert users the performance can be modeled by excluding the RT term and retaining the MT mean term, where RT is the choice reaction time modeled with the Hick - Hyman law. MT mean is computed with digraph probability and the movement time where the movement time between 2 keys is modeled with the Fitts' law. This model is applicable only for the typing behavior where the typist is holding the mobile device in one hand and using a single finger to type.

Now let us consider the case for two - thumb typing, in the earlier model we considered the case for single fingered typing. Now in the case of two thumb typing the behavior is different so; obviously, the model will be different and we cannot use the earlier model in the letter case.

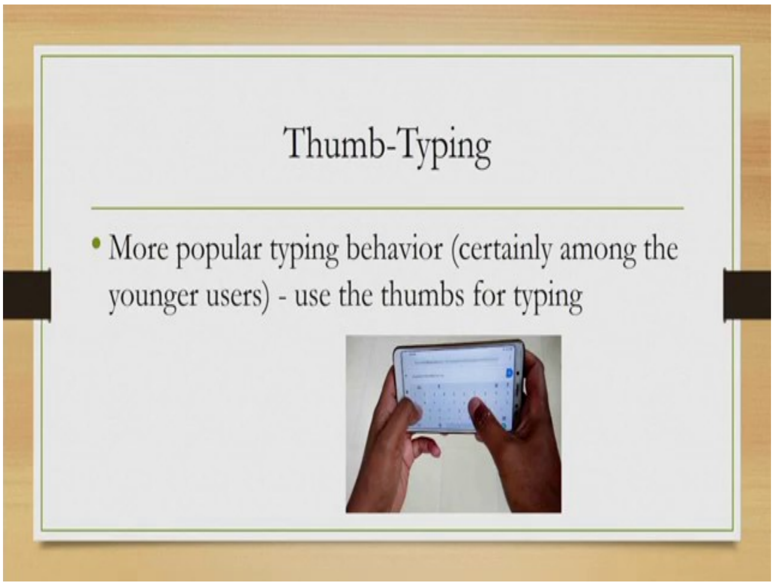
(Refer Slide Time: 25:20)



So, when we talked of single fingered typing you might have wondered in which situations typically this type of behavior occurs when do we type with single finger actually that was the case in the early generation of mobile devices when there used to be stylus and you where forces to type with stylus.

In certain age groups particularly age group with higher age still such type of behavior is prevalent; however, with younger generation more common typing behavior is two thumb typing. So, this typing behavior should be modeled as it is useful in a wide scenario. So, we need to model these two thumb typing because a large number of users nowadays behave in a similar way and the one finger typing model is not applicable for that large group of users.

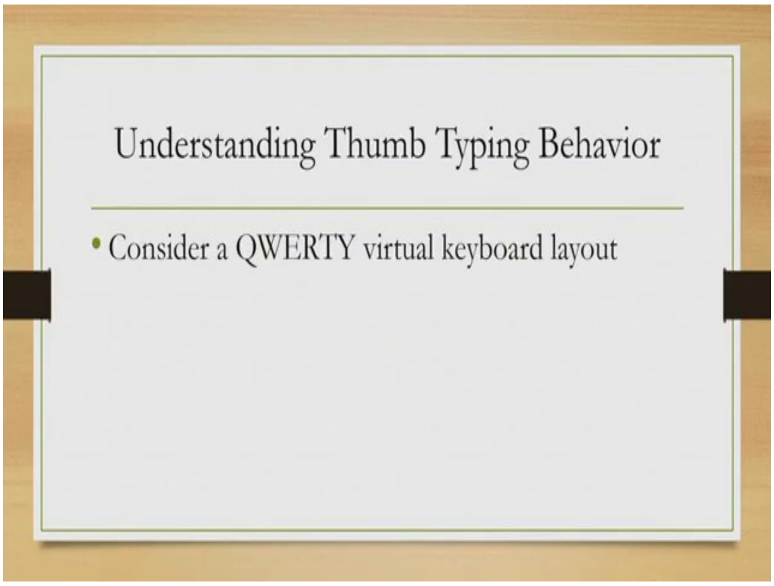
(Refer Slide Time: 26:28)



So, let us try to have a look at the scenario. So, here as you can see the 2 thumbs are used to type and both the hands are used to hold the phone. So, the way the thumb moves is different than the way we used to move the finger in the case of single finger typing.

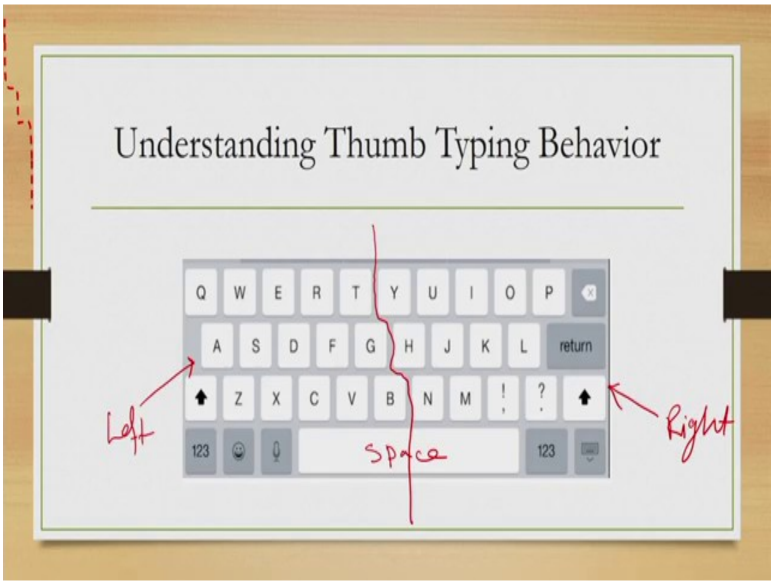
So, accordingly the modeling approach and the overall model will change. Now in order to understand the model we need to understand the behavior when we type using our thumbs how do we behave.

(Refer Slide Time: 27:00)



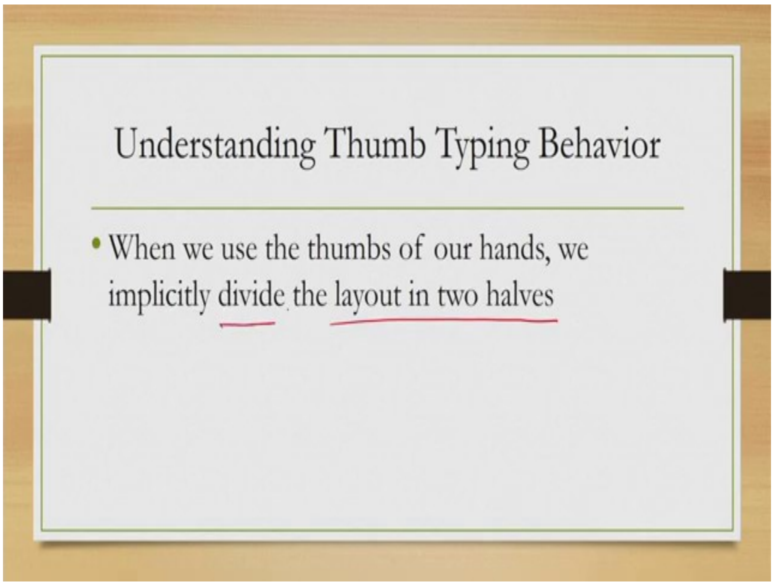
Now, in order to understand the behavior we will use the case of a QWERTY layout, we will assume that the typing is taking place with a QWERTY layout, later on we will talk about generalizability of the model and the idea.

(Refer Slide Time: 27:15)



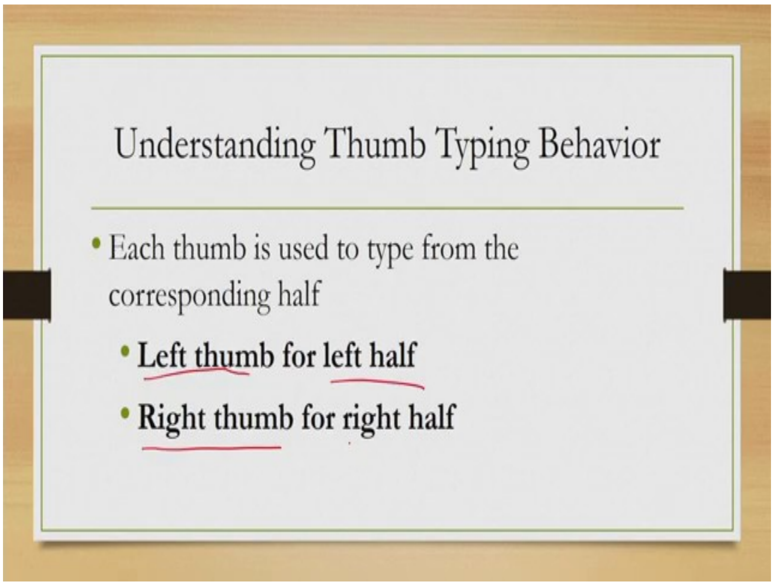
So, in case of QWERTY layout it looks something like this and all of you are probably familiar with this layout. So, mostly we get to see this layout when we type in English in our phones.

(Refer Slide Time: 27:30)



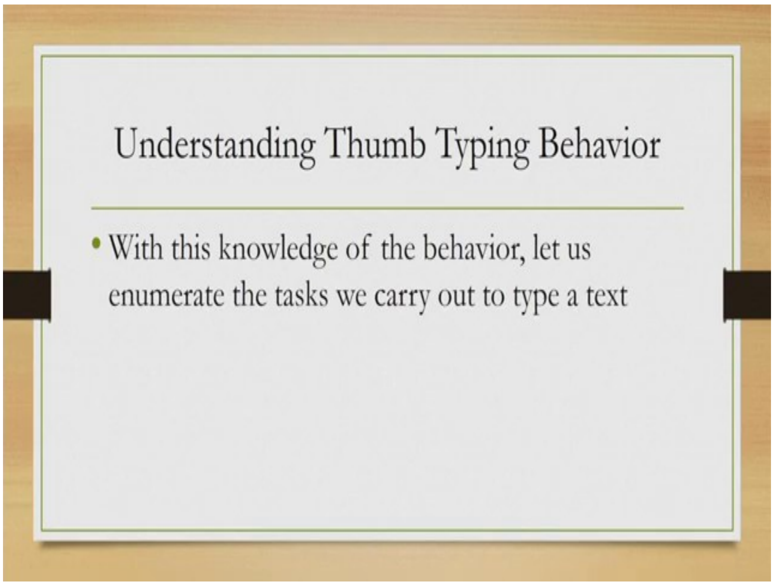
Now, when we use 2 thumbs to type with this layout what happens, we implicitly divide the layout into 2 halves. So, although there is a single layout we implicitly in our mind divided into 2 halves and this division of course, is not easy to figure out, but roughly we can say that if this is the layout somewhere here the boundary lies. So, the this side is the left half when we type and this side is the right half you may not hear that the space key that is here is actually spanning over both the halves. So, that is one typical characteristics of a QWERTY layout that it has a wide space key compared to other keys and even if we are implicitly dividing the layout into 2 halves the space key spans both the halves.

(Refer Slide Time: 28:42)



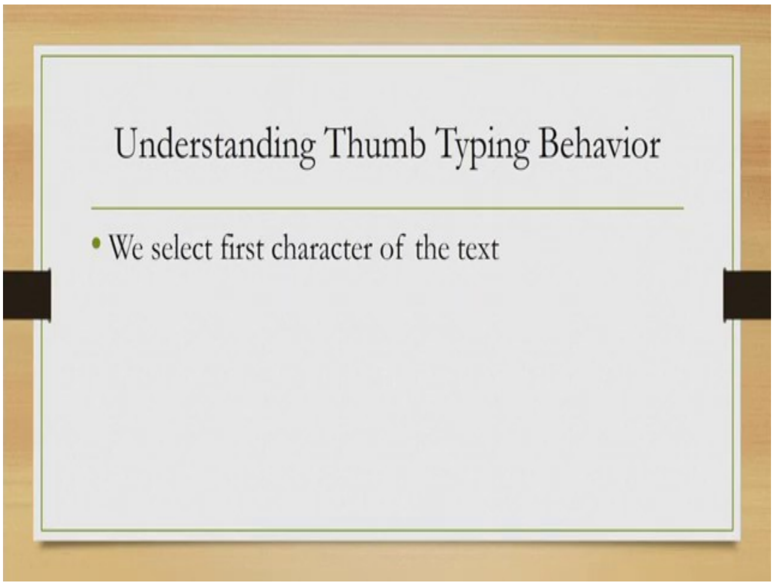
Now, when we type we actually use one thumb to type in one half only. So, we use the left thumb to type in the left half and the right thumb to type in the right half.

(Refer Slide Time: 28:57)



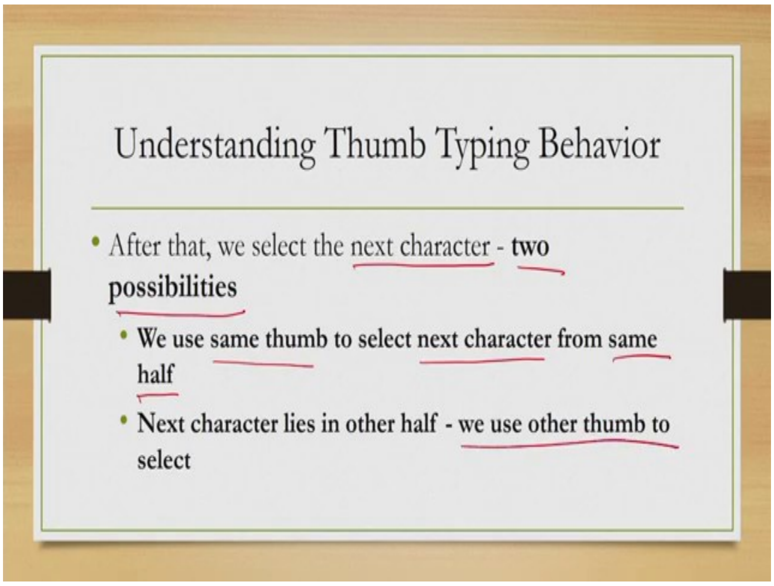
So, with this knowledge of the behavior we can now proceed to understand how to model this behavior.

(Refer Slide Time: 29:07)



So, what are the tasks that we carry out when we are trying to type using our 2 thumbs.

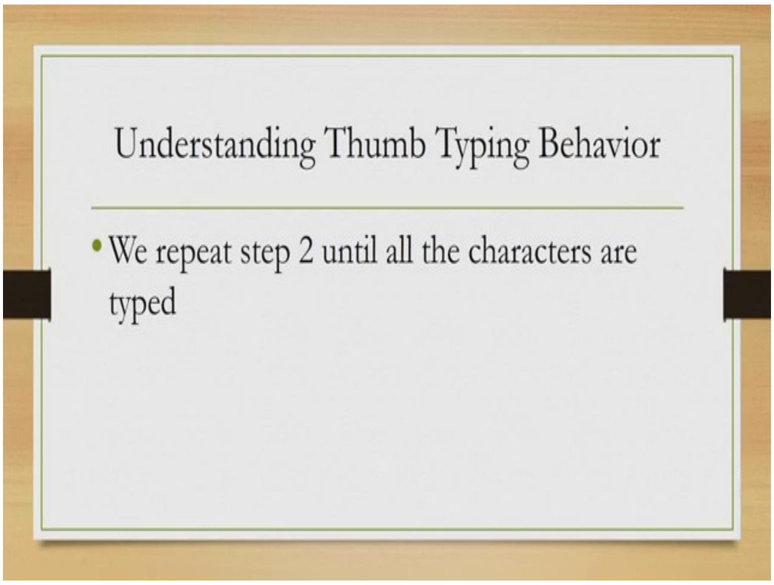
(Refer Slide Time: 29:19)



So, first we select the first character, after that we select the next character, now when we select the next character there can be 2 possibilities. One possibility is that we use the same thumb to select the next character from the same half. So, the previous character was suppose selected from the right half and the next character also falls in the right half. So, in that case so, we use the same thumb that is the right thumb to select both the character the previous character has well has the next character.

Otherwise suppose the previous character was selected in the right half and the next character falls in the left half in that case we need to switch the thumbs. So, a previous character we selected with the right thumb now we need the left thumb to select the next character so, we need to use the other thumb.

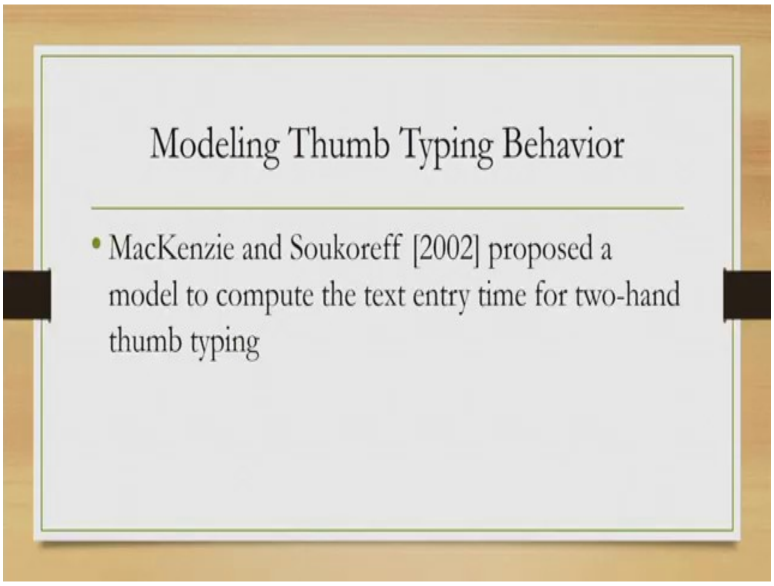
(Refer Slide Time: 30:10)



And the step 2 that is selection of the next character continues with both these possibilities for all the remaining characters in the entire text till all the characters are typed. So, that is in a nutshell how we type just to repeat. So, when we type we implicitly assume that the entire layout is divided into 2 halves, the left half and the right half, each thumb is assigned to one half.

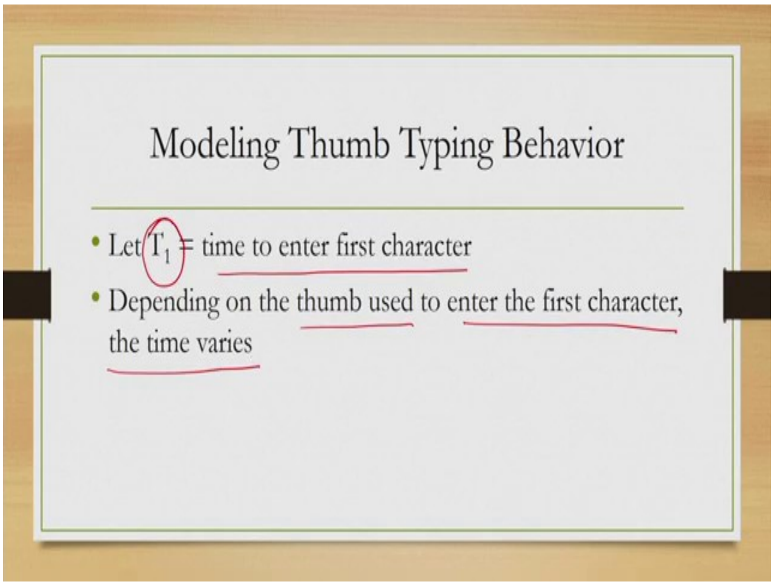
So, characters falling into that particular half is typed with the same thumb. So, left half characters are typed with the left thumb, right half characters are typed with the right thumb. And when we type we enter the first character then when we enter the next character then there are 2 possibilities either the next character falls in the same half in which the previous character was and so we use the same thumb. Otherwise the next character falls in the other half and we use the other thumb to type it and this process continues till we enter all the characters in the text.

(Refer Slide Time: 31:18)



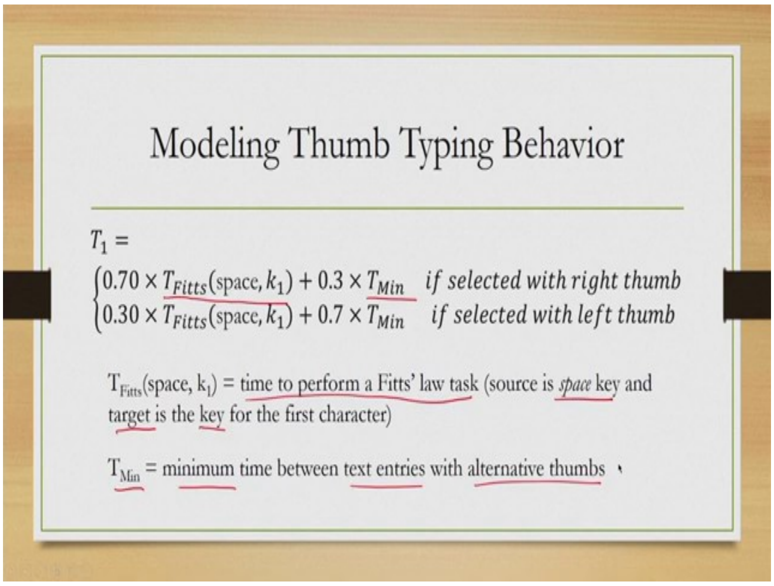
So, then how do we model this behavior? One model was proposed by Mackenzie and Soukoreff in 2002.

(Refer Slide Time: 31:22)



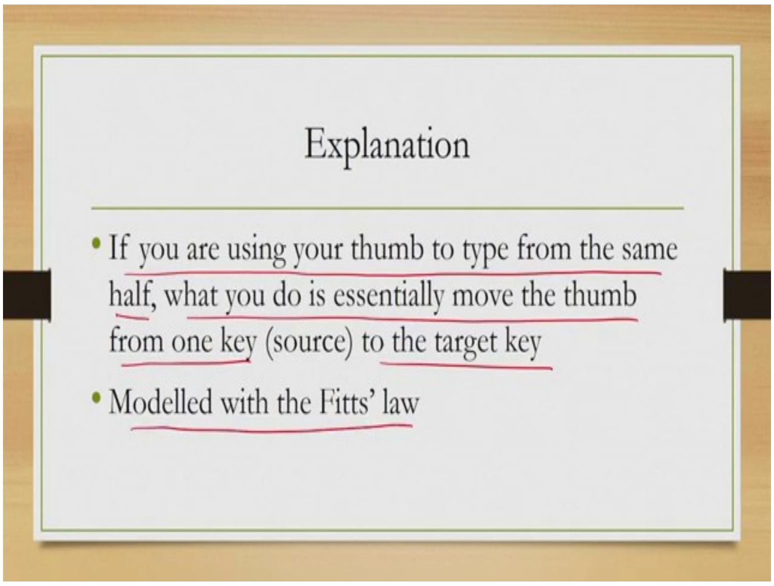
So, what they proposed is that a recursive equation to model the performance of two thumb typing. So, let us try to understand the model. First thing is we first type the first character now let us denote this time to enter the first character by the symbol T 1. Now depending on the thumb used to enter the first character this time varies. So, T 1 is not constant it varies depending on the thumb that you are using to enter the first character.

(Refer Slide Time: 32:04)



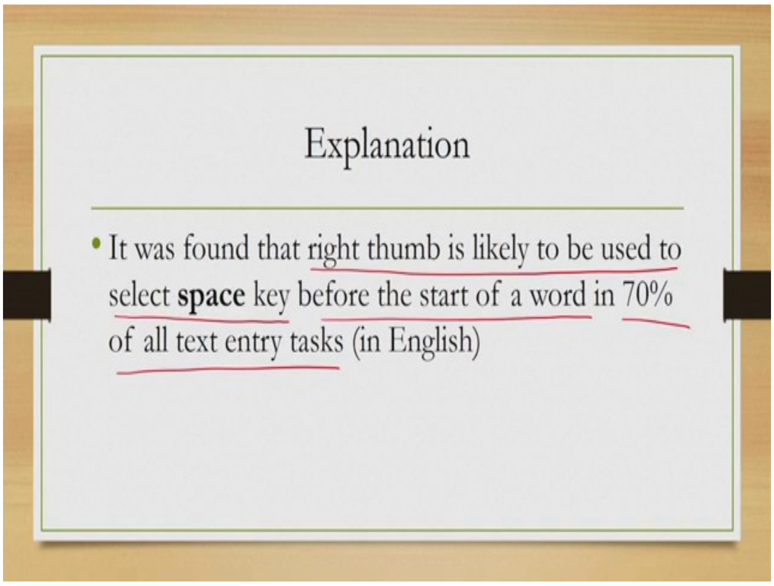
And Mackenzie and Soukoreff proposed a formulation of T 1 to model or to compute this time. So, according to their formulation T 1 can be represented with this equation. So, there are 2 components T Fitts' and T min, T Fitts' represents the time to perform a Fitts' law task that is pointing task or a movement task from the space key to the target key for the first character. And T min is the minimum time required to move the thumb or minimum time required to move between text entries with alternative thumbs.

(Refer Slide Time: 32:51)



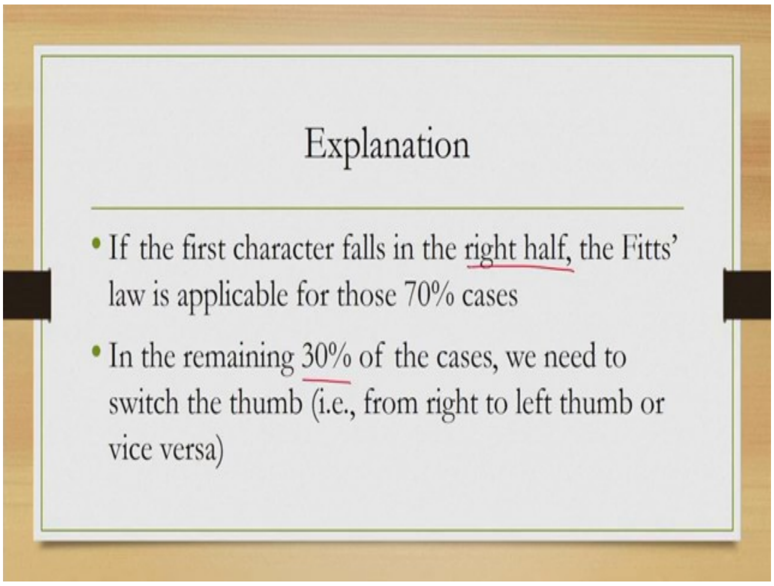
Now, suppose if you are using your thumb to type from the same half, what you do is essentially move the thumb from one key to the target key. Now this movement as is obvious can be modeled with the Fitts' law.

(Refer Slide Time: 33:12)



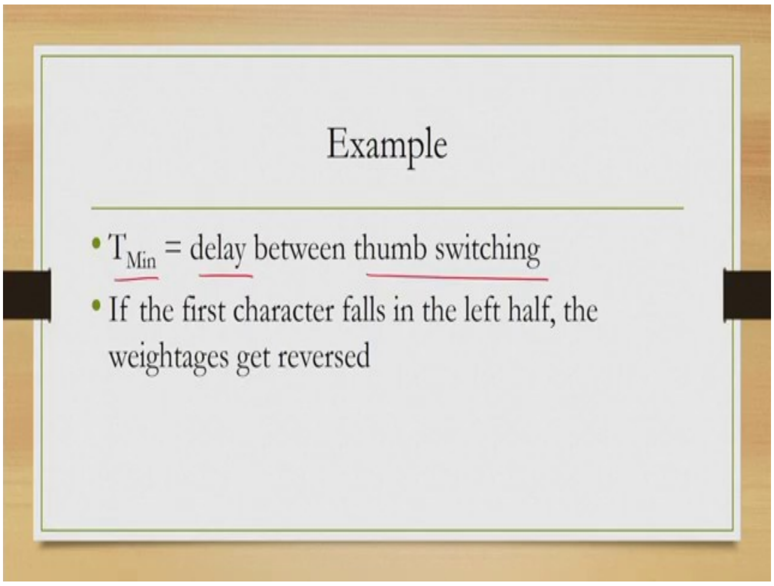
What Mackenzie and soukoreff have found empirical is that the right thumb is likely to be used to select space key before the start of a word in 70 percent of all the text entry task in English. So, that is why this T Fitts' space and the target key that component was there that before you start typing a character it is likely that you have selected the space key and in 70 percent of the cases you have selected the space key with the right thumb.

(Refer Slide Time: 33:53)



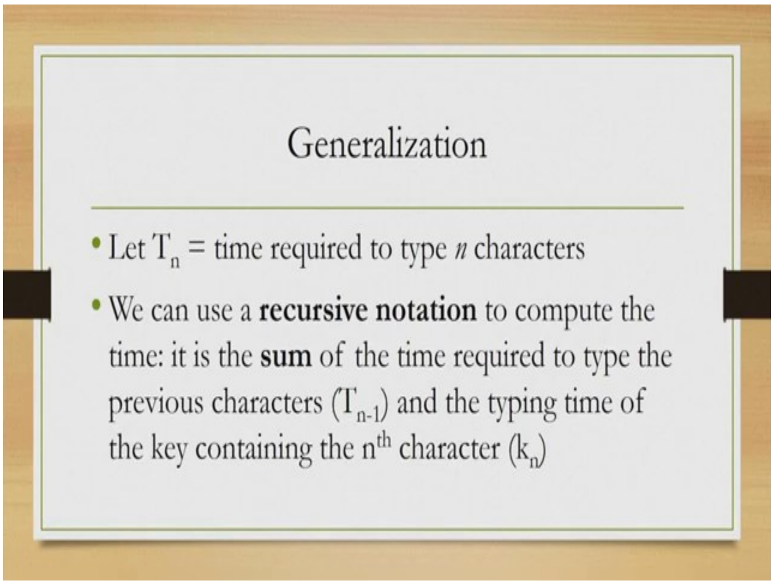
Now, if the first character falls in the right half in that case you have to simply performing a Fitts' movement task that is moving the thumb from the space key to that particular character which is going to be 70 percent of the cases. And in the remaining 30 percent of the cases we need to switch the thumb so, because it is likely to be falling in the other half than the particular thumb. So, in that case what we need is switching the thumb from right to left thumb or from left to right thumb.

(Refer Slide Time: 34:27)



And now T min is essentially the time to switch the thumb. So, if you are typing with the right thumb and now you need to type with the left thumb then the time required to switch or activate the left thumb is represented by T min. Now in case the first character falls in the left half then all this weights get reversed as shown in the equation.

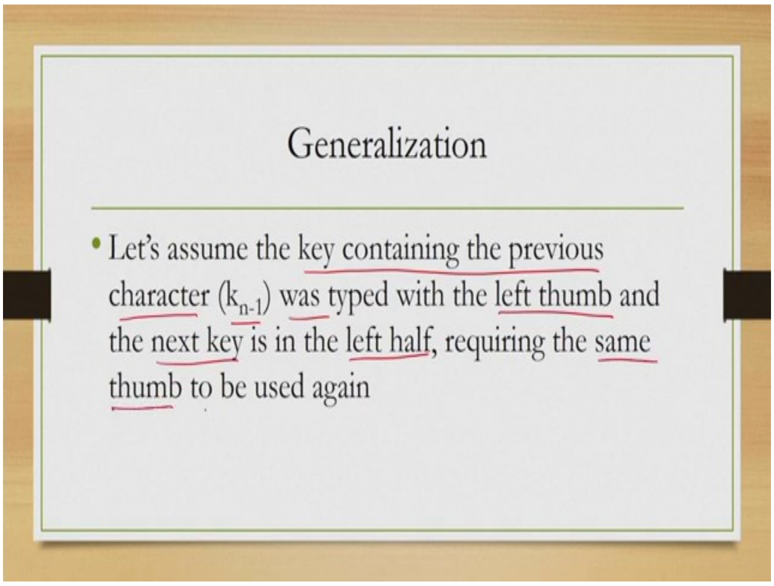
(Refer Slide Time: 34:57)



So, that is about typing the first character then we keep on typing the next characters as we have discussed earlier.

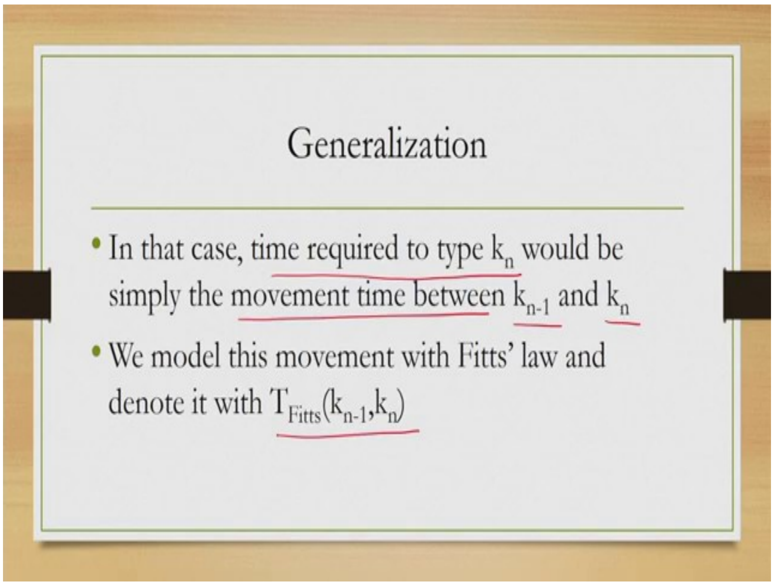
Now, let us denote by T n the time to type up to n characters, now we can represent this T n in a recursive form by summing up the time required to type up to n minus 1 character and the time required to type the n th character.

(Refer Slide Time: 35:31)



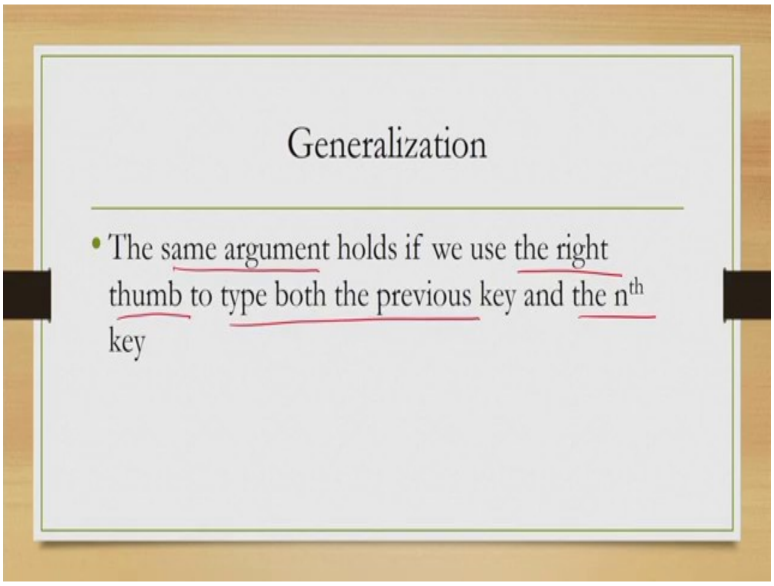
So, let us try to formulate this recursive equation, let us assume that the key containing the previous character that is kn - 1 was typed with the left thumb and the next key is in the left half itself. So, what you need is the same thumb to type that next character because it is in the same half.

(Refer Slide Time: 36:01)



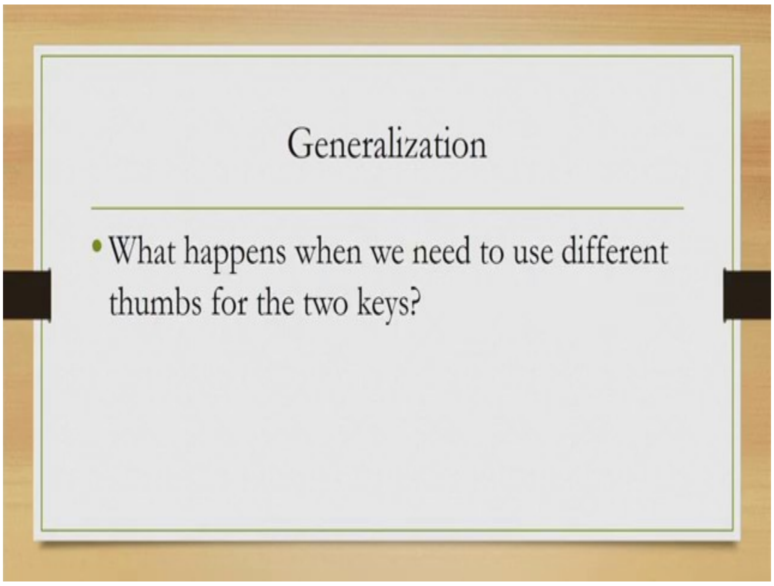
So, in that case time required to type the n th character kn would be simply the movement time between kn - 1 and kn. So, between the previous character and the next character and this movement you can model with the Fitts' law as is pretty obvious.

(Refer Slide Time: 36:20)



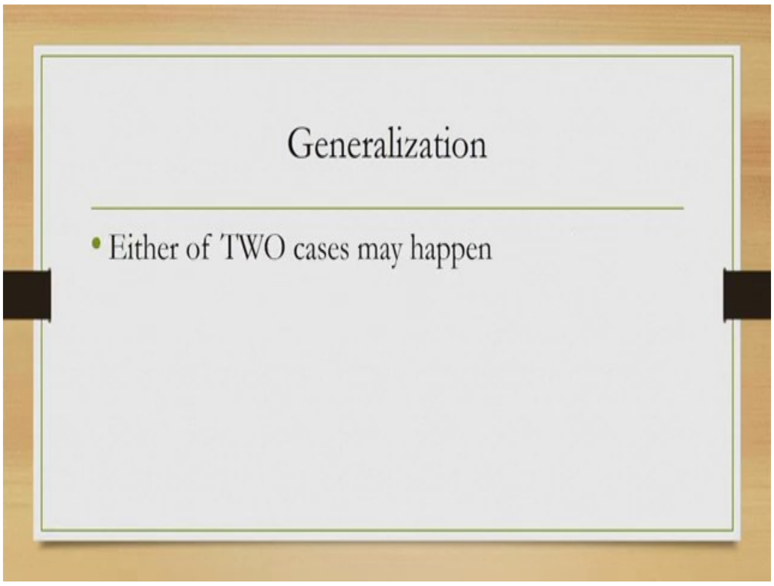
We can have the same argument if we assume that the right thumb was used to type both the previous and the n th key. So, n - 1 and the n th keys were typed both with the right thumb, then again we can simply use the Fitts' law to model this finger movement or thumb movement between these 2 keys.

(Refer Slide Time: 36:48)



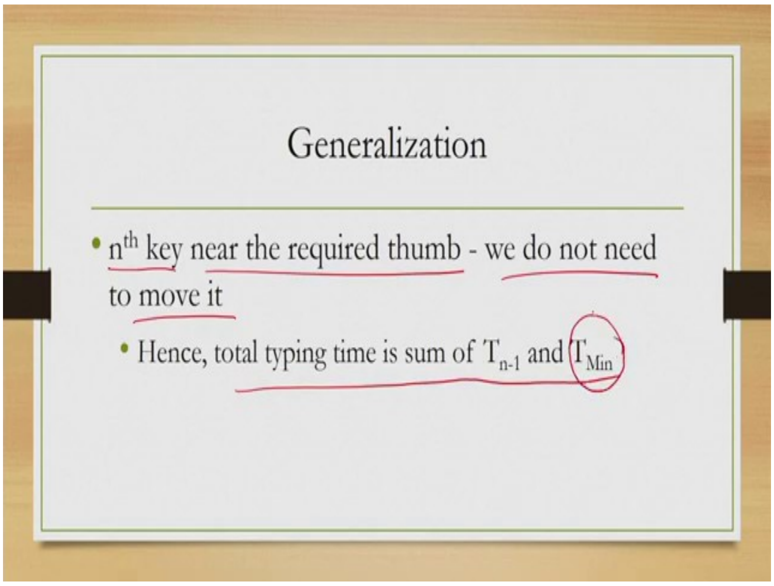
The challenge is or the problem is the scenario where we need to use different thumb to type these 2 characters the n minus 1 th character and the n th character; obviously, they will be in the different halves and we need to switch thumbs.

(Refer Slide Time: 37:10)



So, how to model that? So, there can be 2 cases.

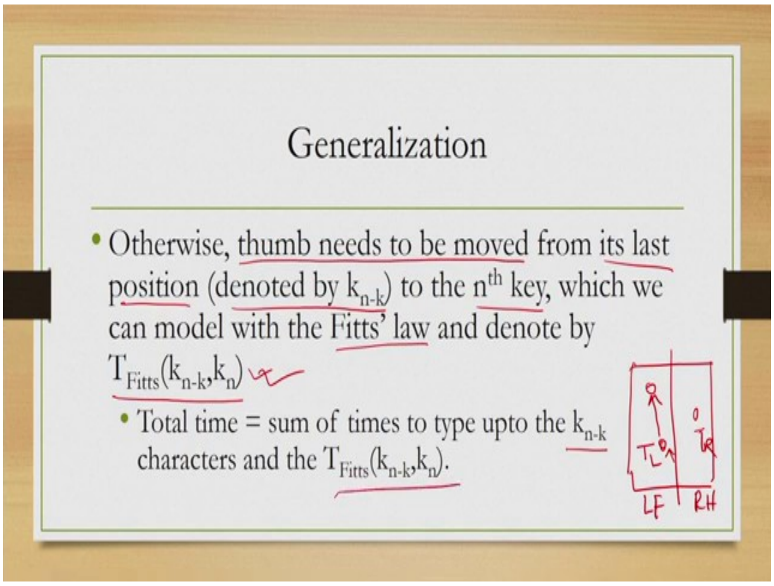
(Refer Slide Time: 37:14)



The n th key near the required thumb. So, the n th key is present near the required thumb; that means, if we have already typed with the right thumb and the next key we need to type with the left thumb and the key is near the resting place of the left thumb then we do not need to move it. In that case we just need to take into account that thumb switching time because the desired thumb is already near the key there is no need to move the thumb and we need only the T min to consider here that is the thumb switching time.

So, essentially the total typing time is the typing time up to T n minus 1 and the thumb switching time the sum of these 2 quantities

(Refer Slide Time: 38:07)



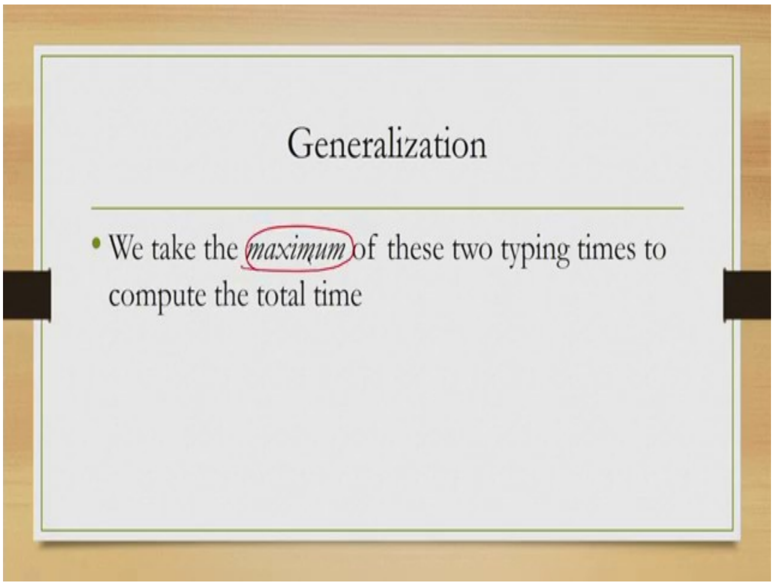
However if the thumb needs to be moved from it is last position which we may denote by the term k n minus k to the n th key then we need to deal with it separately we need to model this movement. Here also we can use the Fitts' law and we can model this movement with simply the Fitts' law as T Fitts' between the 2 keys.

Now let us try to understand this thing with an example. Suppose you are typing 5 character and last character was typed with your right thumb before that you typed with your left thumb and this particular left thumb is now resting at a particular location in the left half. Now the next character that you need to type is falling in the left half now we need to switch the thumb, but this character is not near the resting place of the left thumb that is the same place where you have selected the last character when you used the left thumb pictorially.

Let us assume that this is the half now my thumb is here T left where I have selected a particular character some time ago then currently I have selected from right half, this is the right half, this is left half. Now my right thumb is here then my next character is in the left half and I need to switch and then make a movement because my thumb rests at this place. So, I need to make a movement from here to here.

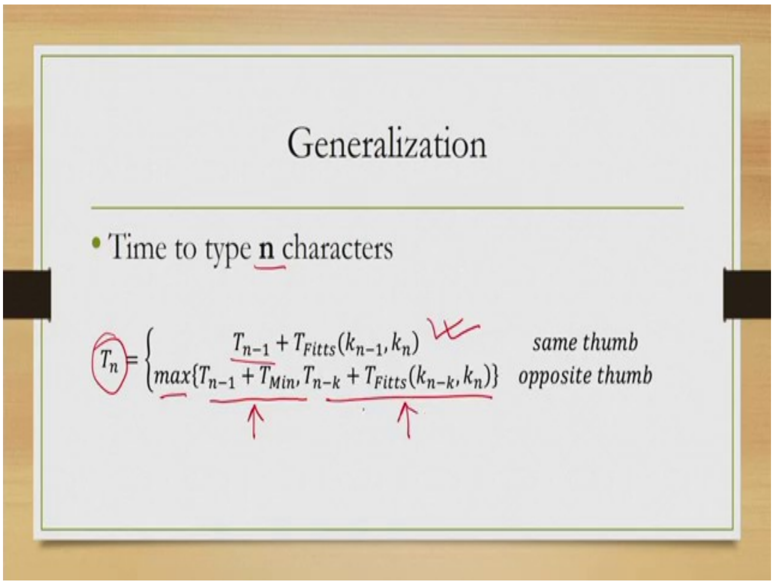
So, this movement we are modeling with the Fitts' law which is shown here. So, then the total time may be some of the times to type up to the k n minus character and the T Fitts' model the time predicted by this movement model.

(Refer Slide Time: 40:22)



Now, both the cases are possible. So, in order to model we take the maximum of these 2 typing times to compute the total time.

(Refer Slide Time: 40:32)

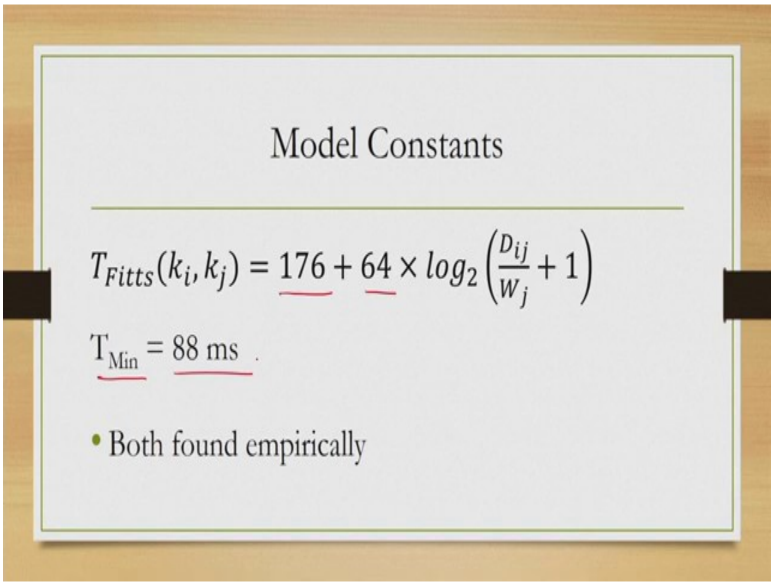


So, then how it looks I want to type n characters then my overall model to compute the typing time can be represented in this way if there in the same thumb that is the n minus

1 th and the n th characters are in the same thumb then we can use this expression. And in case of opposite thumb we take the maximum of these 2 quantities and I have already explained what these quantities mean.

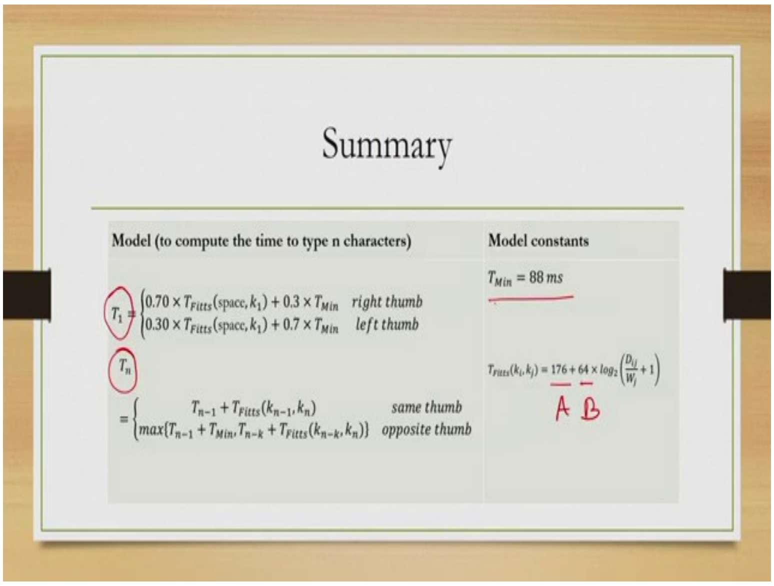
So, when the other thumb that I need to use to type is near the desired key then this is the expression to calculate that and when the other thumb is not near the desired key then this is the expression to calculate that and we take maximum of these 2 when we need to use the opposite thumb.

(Refer Slide Time: 41:32)



And Soukoreff and Mackenzie the authors of that paper also empirically estimated the constant values required to use the Fitts' law model and the value for T min. So, in the context of two thumb typing there empirically estimated that the Fitts' law model constants are 176 for A and 64 for B and the T min the thumb switching time is 88 milli second.

(Refer Slide Time: 42:09)

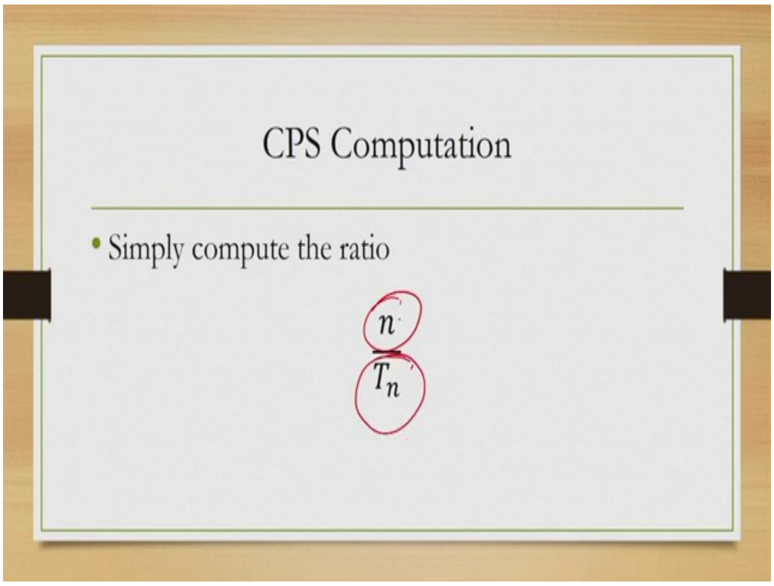


So, we can use these to represent the performance model for true thumb typing. So, that is the overall model. So, let us try to summarize what we have discussed. So, there are 2 components in the model, first is the model of time to type the first character and the second part is the recursive part that is time to type the n characters which can be represented in terms of up to the previous character and the current character and this time to type the first character acts as a boundary value to stop the recursion.

And there are several cases as we have discussed both in the case of modeling first character typing time as well as modeling n character typing times and these cases taken to account the position of the thumb in the particular instant of typing and empirically the model constants are also found T min value is 88 milli second and Fitts' law constants are 176 and 64 respectively for A and B that 2 constants present in the Fitts' law Fitts law modeling of movement time.

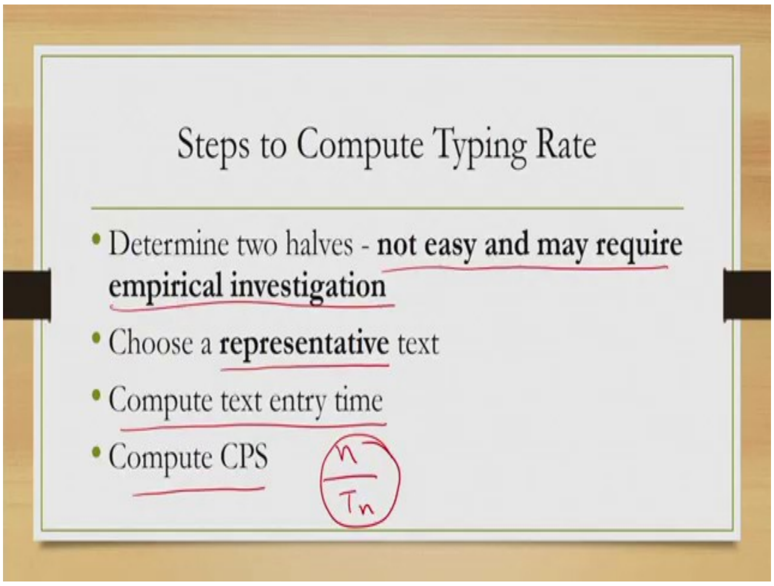
Now this model is of course, not a model of typing performance in terms of text entry rate what it models is essentially the typing time how much time it takes to enter n character. So, then what we need is a small last step that is we need to come up with a model to compute the typing performance based on the model of typing time.

(Refer Slide Time: 44:07)



So, what we can do is simply take the ratio of n by T n, T n is the time to type n characters to get the typing speed in characters per second.

(Refer Slide Time: 44:18)

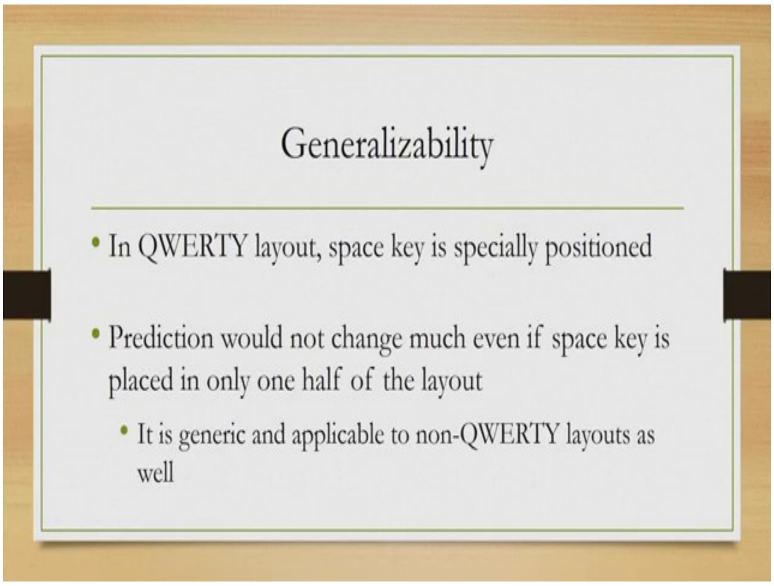


So, essentially what are the steps to compute the typing rate as representative of the typing performance?

So, first we need to determine the 2 halves of the layout. So, we are given a layout and we need to compute it is typing performance in terms of typing rate that an user is expected to have with this layout. So, first thing is we need to determine the 2 halves note that this is not easy and may require empirical investigation. So, you need to investigate empirically how the thumbs are moving so that you can identify the 2 halves so, that is the first step. The next step is we have to choose a representative text now that is important again. So, any text will not work so, you have to choose a text that typically user is likely to type with this particular layout and again there are many ways to find out representative text from a corpus for that particular language.

Once the representative text is found you compute the text entry time using the two thumb typing time model that we have discussed and with that time you compute CPS by taking the ratio as we have shown n by T n. So, that is the model of typing behavior for two thumb typing on virtual keyboard shown on a mobile screen.

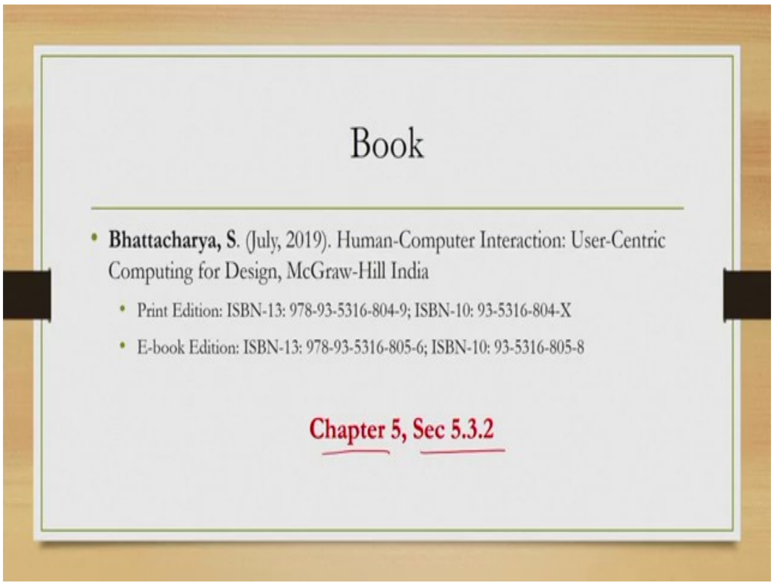
(Refer Slide Time: 45:51)



Now, the last question that we should try to answer is that we have derived the model in terms of an example QWERTY scenario where we assume that the space key is standing on both the halves. Now there are other layouts that may not satisfy this condition space key may be displayed differently. So, whether this particular model is applicable for such layouts has well. The fact is that even if the space key is not spanning over both the layouts that has been found that the same model we can use to compute the overall typing time per n characters and so, this model is generalizable it is not specific to the QWERTY layout

So, before we wrap up I would just like to recollect what we have learnt in today's lecture. So, we have learned about 2 typing performance models by typing performance we refer to the typing rate represented in the form of characters per second or words per minute. And the first model that we have learned about is the Fitts' digraph model which is a model of typing performance for single finger typing on a mobile device. The second model that we have learned about is two thumb typing model which is more popular nowadays, both the models made use of Fitts' law and Fitts' Digraph model made use of Hick -Hyman law in addition to the Fitts' law.

(Refer Slide Time: 47:25)



The material that I have covered in today's lecture can be found in this book, you are advised to refer to chapter 5 section 5.3.2 to get more details on the topic that we have covered today.

Thank you and good bye.