



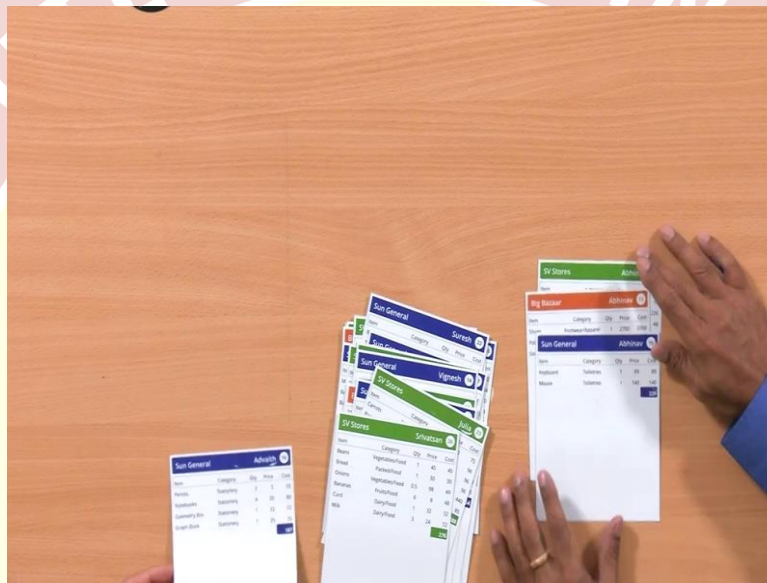
IIT Madras

ONLINE DEGREE

Computational Thinking
Prof. Madhavan Mukund
Prof. G. Venkatesh
Department of Computer Science
Chennai Mathematical Institute
Indian Institute of Technology, Madras

Lecture - 4.4
Concept of binning to avoid the complexity of nested iterations

(Refer Slide Time: 00:15)



So, let us move to the shopping data.

So, you know one of the things that we might want to ask is where people are shopping? So, do they kind of visit different shops for the same type of things, do they go to different shops?

Some people do that right. They go from one shop to another; they want to compare prices.

They are always.

But comparing prices does not mean that they will buy.

Correct.

They may be compare, and they come back and buy in the same place.

Yeah.

And in fact, shops actually try to keep their customers coming back to them right. So, they give them these bonuses and loyalty points, and all these kind of things. So, maybe that is a good idea. Let us try to see if we were a shop for instance right, how would we try to?

So, what?

Look for.

So, you are saying basically there are some customers who will who do not like to buy everything from one shop, they will keep going to different shops and buying things.

Yeah.

They may buy grocery for example from one shop, and there are some other people who will go to Big Bazaar or something and they will buy everything from there right.

Yes.

They do not want to go to many shops.

Correct.

They want to save their time; they go to one shop.

Yeah.

And they buy everything (Refer Time: 01:15).

Thinking.

So, can we identify, can we identify whether there are such customers here who go to only?

Do you want to call them something I mean these customers.

Let us call them loyal, because this is.

Loyal to shops.

Loyal to shops well.

Loyal to shop.

So, loyal customer.

So, loyal customer, a loyal customer is a customer who firstly so he should have visited he should have visited many times.

Yeah.

But otherwise this question does not come right. He should have visited many times, but all those visits are to the same shop.

Correct. So, it cannot be just one bill.

So, yes, basically generating many bills, there are many bills with the customers' name on it.

Yeah.

But all these bills are with the same shop on it.

Yes.

That such a customer is a.

Loyal customer.

Loyal customer ok, that is fine.

So, what we need to do? Let us do it.

What we do that so?

Most obvious way I can think of.

Is that we take a bill right which has a name of a customer.

Customer.

And the name of the shop on it. And now we try to see all the other bills and see if there is (Refer Time: 02:12)

For the same customer.

with the same.

Customer.

Same customer with a different.

Different shop.

So, we have to keep like we did, it is an iteration.

So, we are iterating through this. We have picked one card.

We take the name from the card. Then we iterate through all the cards again.

Well.

Nested iteration right.

Yes.

It is a nested iteration.

But we can maybe do some collection, because it will be because when we are collecting we do not need to compare this customer with other customers right.

So, it is filtering. So, you are filtering on this customer.

So, we can filter on this on all the bills of this customer, and then check that group. So, maybe we are doing some kind of binning as we are going along by customer

So, if you bin first?

Maybe we should bin first.

Or bin as we go along?

I think, we bin as we go along I think. So, we bin for the first customer.

Rather than first trying to bin because.

Yeah rather than.

This might be faster right.

So we.

Bin as we go along

ok.

So we just.

So, what we have done, we have taken Abhinav now what we want to do is collect all the Abhinav cards

Correct.

Right.

we see a customer and say this is Advaith, Rajesh, Akhil, Ahmed, Srivatsan , Radha, Vignesh, Sudeep, Neeraja, Suresh, Aparna, Ahmed, Srivatsan , Akshaya, then we have Abhinav.

Alright.

So, in particular, we have seen now that there are.

Abhinav has two shops; one is Big Bazaar and one is SV stores.

So we already know that he is not a loyal.

Abhinav is not a loyal customer (Refer Time: 03:35) .

But let me first pick out all the Abhinav's

because

Because again Abhinav will come back.

And also once we have seen Abhinav, we do not need to see Abhinav again know.

Exactly.

Because.

So, let us finish this iteration.

Let us finish the Abhinav

And make sure there are no more Abhinav

Let us finish off Abhinav.

So, Neerja, Akshaya, Akshaya, Srivatsan , Advait, Ahmed, Julia, Srivatsan .

See one more Abhinav.

Ok.

And another is here

So, he is really not loyal; he went to all the shops ok.

Yeah.

So, we.

So, we have three different shops.

So, Abhinav is gone.

So, now, we have Advait.

(Refer Slide Time: 04:09)



So, we do the same thing. So, we look for all Advait's in this thing. So, we have one more Advait,

So, this is a nested iteration. Actually we are nesting; it is a kind of a nested iteration.

Correct, and as we are nesting,

but

we are eliminating.

But, taking away a large number of cards.

So, not just the one card we are using in the outer iteration, we are taking away all the cards.

All the cards.

With the same name.

With the same name.

On the (Refer Time: 04:33).

So, ok.

So, it is a little bit.

Its moving faster.

Hopefully.

Right, see.

So, I do not see any more Advaitis there it is

One more Advait's.

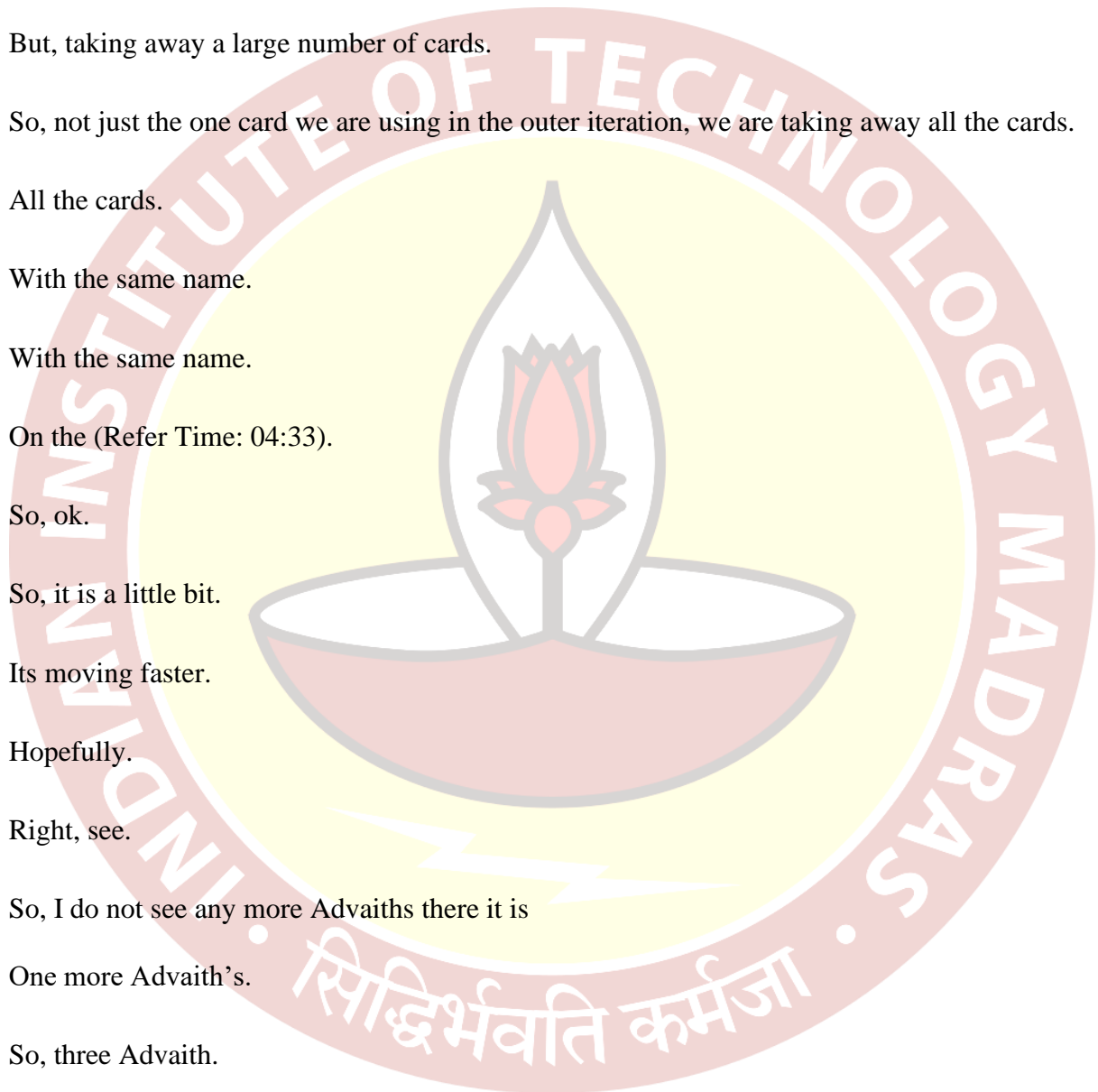
So, three Advait.

Again Advait has gone to all the three shops.

So, definitely not.

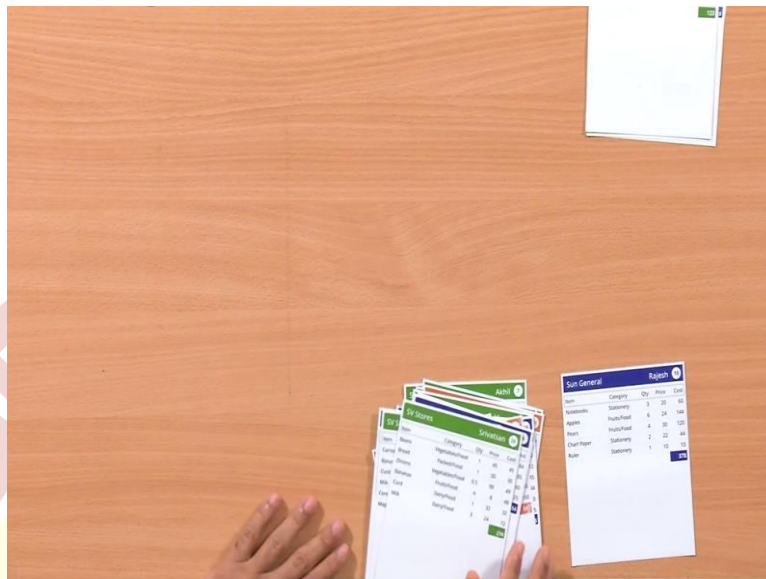
So, he is also not loyal.

Not loyal



So, he is also gone ok.

(Refer Slide Time: 04:53)



Now, we have Rajesh, alright, Akhil, Ahmed, Srivatsan , Radha, Vignesh, Sudeep, Neerja, Suresh, Aparna, Ahmed, Srivatsan , Akshaya, Vignesh, Mohith Ahmed, George, Neeraja, Akshaya, Akshaya, Srivatsan , Ahmed, Julia, Srivatsan ,.

So, only one.

So, here we say is the.

so we don't need them

We said that we cannot.

Yes.

Even determine whether he is loyal or not.

Because he has only made one visit

one visit

to a shops so

So, that is.

Too little data, too little data to decide.

Correct.

So, again I will reject it.

I think we should,

Ok

Does not meet our criterion.

Ok.

Srivatsan ,

(Refer Slide Time: 05:27)



Julia, Ahmed, Srivatsan , not loyal,

Ok

but we have to find the others, if there are any more.

It does not look like anybody is loyal.

Maybe everybody is just looking for the best bargain and going to the shops.

Which suit them? Here is one more Srivatsan,

Ok

So, srivatsan (Refer Time: 05:50) two shops.

Two shops.

But Two shops.

He did not go to Big Bazaar.

Yeah.

He went to SV stores and Sun General alright ok.

(Refer Slide Time: 05:57)



Now, we have Akhil.

Ahmed, Radha, Vignesh, Sudeep, Neeraja, Suresh, Aparna, Ahmed, Akshaya, Vignesh, Mohith, Ahmed, George, Neeraja, Akshaya, Akshaya, Ahmed, Julia. So, no because it is only single card.

Single card.

So, no data to decide.

Yeah.

Right.

(Refer Slide Time: 06:23)



Similarly, Julia, so let us see Ahmed, Akshay, Akshay, Neeraja, George, Ahmed, Mohith, Vignesh, Akshaya, Ahmed, Aparna, Suresh, Neeraja, Sudeep, Vignesh, Radha, Ahmed.

Again one.

Singleton.

Singleton.

Ahmed.

(Refer Slide Time: 06:41)



Radha, Vignesh, Sudeep, Neeraja, Suresh, Aparna, Ahmed, so far he is loyal. Akshaya, Vignesh, Mohith, Ahmed, seems to be loyal. Goerge, Neeraja, Akshaya, Akshaya, a last.

So, Ahmed was almost a loyal customer until the fourth bill came and changed ok.

So, Ahmed is.

See you know one thing I noticed basically is that when you are doing this right, we keep coming to these Akshaya, all these people again and again. So, if you had binned them.

(Refer Slide Time: 07:11)



Maybe, we should have binned them to begin with that is also possible.

Shall we will try now? We will list (Refer Time: 07:20).

Let us try that ok. Let us try a binning.

Keep on binning them actually Akshaya,

Akshaya, but then we have to keep track of bins which we can do of course, Akshaya.

Then Neeraja.

Then George and Mohith, Vignesh, Akshaya, Aparna is a new bin. Suresh, Neeraja, Sudeep, Vignesh, and Radha. So, these are the remaining customers.

So, this is a singleton.

Singleton is out.

Singleton.

Singleton is out.

Out.

Aparna is a singleton that is out; Sudeep is a singleton that is out.

Out.

George is a singleton that is out.

Suresh is a singleton that is out.

And now for each of these, we just have to go through it once and check how many shops are there.

Yeah.

Right. So, here we can see one different one. So, there are two shops.

Two shops so this is gone.

So, this is gone.

That is gone here again there are again two shops.

Two shops.

Is gone

Gone

(Refer Slide Time: 08:17)



And finally, we are left with one customer.

Who is loyal?

Neeraja, who has gone to two shops and is loyal. Maybe she if she had gone to a third shop, she would not have been loyal, but at least

she is

so far she is .

Two, two visits, two visits.

She is going two visits, yes.

Two bills, all of them from the same shop. So, we found one loyal customer Neeraja.

One ,One, one loyal customer.

Only one loyal customer ok.

So, here it looks like.

So, here what is what kind of problem is this?

Yeah.

This is.

So, we needed to find a customer's bills, and we thought first it will compare the customers' bills with all the other bills, but actually we only need to compare with their own bills.

And then we said that if you are comparing with your own bills, you are actually just grouping it.

Binning.

With customer. So, actually by just binning we create bins one per customer, and then within each bin we only have to do one singleton.

So, there is no need to nest.

There is actually no need to nest.

No need to nest.

No nested iteration at all.

No nested iteration; at all you can go really fast, but you will have to create lot of bins.

Yes.

As many as customers.

Yeah.

So, first separate it by customer.

Yes.

And then for each customer.

Just.

Just go through the list of cards.

And see how many shops are.

So, here there is a problem actually which looked like the earlier problem. So, it looked like there is a nested iterations sitting here right.

But actually there is not one.

But actually there is not one.

Yeah.

So, clever about it you can you can eliminate the nested iteration.

Yeah.

Right.

So, it is it requires some thought then just looking at the statement of the problem and deciding whether you need something or not.

And the reason why we want to knock off this nested iteration is because each time you do nested iteration, it decreases the number of comparisons.

Yes definitely.

Enormously.

Enormously.

Right.

Because.

Enormously.

You just.

Take so much time.

You just multiply the number by itself and you get a huge.

Huge number.

Explosion in the number of operations or comparisons or whatever that.

But there is a trade off I mean in the sense that if you are comparing every card with every card it is counting lot of comparisons are happening, but we had only two piles.

Small space, we are able to do it with this much of desk space.

Yes.

But the minute you.

Put bins yeah you need to.

Whole desk was filled up with cards.

Right.

All the bins coming up on the desk. So, you had much more space you are using.

But you end up doing it much faster.

But much faster.

Yeah so, there is.

So there is some trade off between

Space and the time it takes to.

Time it takes.

So, That is something to keep track off maybe as we get on to larger problems, because right now for us it fits (Refer Time: 10:37).

Desk space.

It is fine.

But if we have more cards and fit on a table top, we would not be able to keep track of the bins.

But the computer I guess memory it can store any number of things no

So, we have to check I mean at what scale we are doing.

Right.

Right. So, supposing we are doing something at the level of population of India then we have.

1.3 billion.

1.3 billion.

So, a computer cannot store everything maybe.

Then bins may be complicated.

Too many bins.

Too many bins.

Alright ok.

