

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias:

Reflecting on CS Graduate Admissions

I was just indulging in my favorite bad habit (reading hacker news), when I came across a discussion about [prestige in faculty hiring](#) - namely, the fact that the majority of CS professors across the US come from the top 18 or so universities, and that even within that, there's bias. The HN discussion turned to Ph.D. admissions, with a commenter noting:

"I'm not sure, but I'd bet that the best institutions actually don't care where their applicants come from. They can afford to just choose the best. It's the so-so places that have to signal their quality by 'hiring the best'. I've seen this in graduate school applications - students with poor marks often stand a better chance of admittance in a top program than a so-so one."

I just finished chairing the CMU CS Ph.D. admissions committee, so I thought it might be worth writing down my experience in doing so. With two very important notes:

1. I'm not doing it next year, so please don't use this as an excuse to drop me an email saying "CAN I GET IN PLEASE" - I'm going to be on sabbatical, in fact, so I'll be about as far removed from the process as possible.
2. No, this was *not* the program that sent out the accidental admissions letters. We dodged that one narrowly (we use the same codebase). phew.

When I ran the committee this year, I gave the committee a few very explicit instructions:

1. Don't spend a lot of time reviewing the top 2% of applicants. They're easy to identify, they get admitted everywhere, and the error rate is low.
2. Don't spend a lot of time reviewing the bottom 50% or differentiating among them.
3. Please focus your attention on finding the students who have the potential to be in the top 10% percent, but are likely to be overlooked by other top programs.
4. Go watch two videos about dodging hidden bias: