01

Slide 1:

Hey everyone, today we're going to discuss how to write a research abstract. You'll learn the essential moves, which are Introduction, Objective, Methodology, Results, and Conclusion. This skill is crucial when summarizing the key points of a research article.

Slide 2:

Let's start with the first two moves. Move 1, the Introduction, is all about giving your audience a brief idea of the topic and why it's important. Move 2, the Objective or Research Question, is where you clearly state the main goal or question that your study aims to address.

Slide 3:

Now, onto Moves 3 and 4. In Move 3, the Methodology, you'll provide an overview of the methods used in your study. It's essential to be concise and informative here. Move 4, the Results, is where you summarize the main findings of your research, giving your audience an idea of what your study discovered or concluded.

Slide 4:

Finally, we have Move 5, the Conclusion and Implications. Here, you'll state the significance of your findings and their potential impact. To wrap up your abstract, use clear and concise language, avoid jargon, and maintain a logical flow from start to finish. This will make it easy for readers to follow and understand the content of your research article.

02

Slide 1:

Hello everyone! Today, we're going to discuss writing an introduction for computer science research articles. We'll cover the essential rhetorical moves, which are introducing the research topic and context, identifying the research gap, stating significance and relevance, presenting the research objective or question, and providing an article outline.

Slide 2:

Let's dive into the first two moves. Move 1 is about introducing the research topic and context. In this step, you'll briefly present the general area of research and set the stage for your specific focus. Move 2 is about identifying the research gap. Here, you'll point out the limitations or unaddressed areas in existing research, justifying the need for your study.

Slide 3:

Now, let's look at Moves 3 and 4. In Move 3, you'll state the significance and relevance of your research. This means explaining why addressing the identified gap is important and how your study contributes to the field. Move 4 is all about presenting your research objective or question. You'll clearly state the main goal or research question that your study aims to address.

Slide 4:

Finally, we have Move 5, which is providing an article outline. This is an optional move, but it can be helpful to give a brief overview of the paper's organization and guide the reader through the subsequent sections. To wrap up, keep these writing tips in mind: use clear and concise language, avoid excessive jargon, and ensure a logical flow between the different rhetorical moves. Thanks for listening, and good luck with your research articles!

03

[Slide 1:

Introduction to the Method Section

Hello everyone! Today, we're going to dive into the world of research articles in computer science and explore an essential section called the method section. It's like the blueprint that helps other researchers understand and replicate the work. So, let's get started!

[Slide 2:

Method Section - Product Creation

Now, let's focus on product creation within the method section. Imagine you have a problem to solve, and you want to create a software product to address it. Here are the key points to consider:

- First, clearly define the problem or objective your product aims to tackle. What is the purpose behind its creation?

- Next, explain the design and development process. Which methodologies or frameworks did you use to guide you along the way?

- It's essential to specify the programming languages, tools, and technologies you employed during the implementation phase.

- Highlight the key features and functionality of your product. What sets it apart from existing solutions?

- Lastly, describe the methods you used for testing and evaluating your product. Did you conduct unit testing, user testing, or evaluate its performance in specific scenarios?

[Slide 3:

Method Section - Experimental/Evaluative Studies

Moving on to experimental and evaluative studies within the method section. Imagine you're conducting research or evaluating existing solutions. Here's what you need to include:

- Start by stating your research question or hypothesis. What are you trying to investigate or prove?

- Describe the participants or data collection process. Who are the individuals involved, and what criteria did you use to select them?

- Specify the variables you studied and the measures you used to collect data. Which metrics or performance indicators did you consider?

- Explain the experimental setup, including the hardware, software, or simulation environment you utilized.

- Detail the data collection procedure. How did you gather the necessary data? Did you conduct experiments, surveys, or interviews?

- Outline the statistical or analytical methods you employed to analyze the data. Which tools or algorithms did you use?

- And don't forget to address any ethical considerations, such as informed consent, data privacy, or institutional review board approval.

[Slide 4:

Summary or Conclusion

To wrap up, let's summarize the key points and importance of the method section:

- The method section serves as a vital part of research articles and product creation, providing a detailed account of the methodology used.

- It's crucial for clarity and reproducibility, allowing other researchers to replicate and validate the study or product.

- By providing a well-written method section, we contribute to advancing knowledge and innovation in computer science.

In conclusion, the method section is like a roadmap for others to follow and build upon our work. It ensures that our research or product creation is transparent, reproducible, and valuable to the field of computer science.

04

Today, we will explore how to write the Results section for a research article, focusing specifically on computer science.

[Slide 1]

Let's start with an overview of the Results section. Its main purpose is to present the primary results of your research. It's all about the facts, without interpretation. The information you include should be clear, concise, and directly answer your research question or questions. You'll want to use both text and illustrative materials like tables and figures to present your findings in a comprehensive manner.

[Slide 2]

Now, let's dive into the key elements of the Results section. First, you need to state your main findings. These are the core results that directly relate to your research question. Next, you'll provide detailed results. This includes further context, specifics, and accurate details to really paint a picture of your findings. And don't forget about visual aids. Graphs, charts, tables, and diagrams can really help illustrate your results in a digestible and understandable format.

[Slide 3]

But we're not done yet. There are a few more critical elements to cover. If you have any negative results, or results that contradict your hypothesis, you should still report them. This is important for maintaining scientific integrity. Then, you'll want to summarize your findings. This is a brief recap of your main points. Lastly, remember to avoid interpretation in the Results section. Your opinions, implications, and interpretations belong in the Discussion section.

[Slide 4]

So, to wrap up, the Results section provides the evidence for your conclusions. It should be clear, concise, and objective, and visual aids can really help to improve clarity. All findings are important, even the negative ones, as they contribute to the integrity of your research. And remember, keep your interpretations for the Discussion section.

Thank you for listening to this presentation. Keep these guidelines in mind when you're writing your own Results section, and you'll be on the right track. Best of luck with your research!

05

\*\*Slide 1: Introduction to the Discussion Section\*\*

Hello everyone, welcome to our discussion on the importance and structure of the discussion section in a research article. Now, why is the discussion section so critical, you might ask? Well, it's the place where you get to interpret your results, highlight the important findings, connect them to other research, and discuss what it all means. It's the heart of your research article, where you answer the 'so what' question about your findings

\*\*Slide 2: Summarize and Interpret Findings\*\*

Moving on, the first step in writing a discussion section is to restate your research question and summarize your main findings. It's like a brief encore of your results section. For instance, you might say, 'Our analysis shows a significant improvement in the speed of sorting algorithms when using our proposed method.' Next, you delve into the meat of the discussion: interpreting these findings. What do these results mean in the context of your research? A simple interpretation might be, 'The reduction in time complexity means our sorting method can handle larger data sets more efficiently, which is great news for big data applications.

\*\*Slide 3: Compare to Prior Work, Discuss Limitations, and Implications\*\*

Now, no research happens in a vacuum. It's time to connect your work to the broader research field. Compare your findings with previous studies. If your work supports prior research, say so. If it diverges, explain why. For instance, 'These results support the findings of Doe et al., but our method differs in that we implemented a hybrid approach...' And let's not forget about limitations. Every study has them, and it's important to acknowledge yours. Maybe you only tested your method on synthetic datasets. That's okay! But it's a limitation that should be mentioned. Finally, end on a forward-looking note. Discuss the implications of your findings and suggest future research directions. For instance, 'Our method could be tested on real-world datasets in future studies.

\*\*Slide 4: Conclusion of the Discussion Section\*\*

To sum up, the discussion section is your chance to explain what your findings mean, how they relate to other research, what their limitations are, and how they can pave the way for future research. It's your opportunity to show your work's significance to the broader field. Thanks for tuning in, and happy researching!

06

Slide 1:

Introduction to the References Section

Hello everyone, we're here to discuss the references section in research articles. This section acknowledges others' work, supports your arguments, and guides readers to your sources. Its format depends on the citation style required by your target journal or conference. The common styles in computer science are IEEE, ACM, and APA. The key here is consistency in formatting.

Slide 2:

Referencing in LaTeX

Now, let's dive into referencing in LaTeX. In LaTeX, whether you're using a .bib file or the \bibitem command, you'll use the `\cite` command for in-text citations.

The .bib file method is like creating your personal library. Each reference gets a unique identifier. At the end of your document, the `\bibliographystyle` and `\bibliography` commands display your references.

The \bibitem command method involves writing each reference directly into your LaTeX document. It's a bit more manual but follows the same citation process.

Slide 3:

Referencing in MS Word

Switching gears to MS Word, there are two main methods.

First, using a reference management tool. This could be Word's built-in tool or others like Mendeley, Zotero, and Endnote. These tools handle storing references, formatting them, and inserting in-text citations.

Second, manual input. Here, adherence to your chosen citation style is crucial. As an example, in the IEEE format, it looks like this: Author Initials. Surname, "Title," Journal Title, vol.#, no.#, pp. page number/s, Month Year.

Slide 4:

Conclusion

To wrap up, the references section is essential for academic integrity, supporting arguments, and guiding further exploration by your readers. Whether using LaTeX or MS Word, maintaining consistency and accuracy in your references is key. It's all about making your work clear, credible, and reproducible. Thank you, and I hope this presentation helps in your research writing journey.

07

Slide 1: Introduction

Hello everyone, today we're going to talk about a key part of any research paper: the Related Works section. This section is where we connect our research to the wider scientific community. It's our chance to show we know the field, and more importantly, how our work fits into the big picture. The process involves identifying our key themes, finding relevant literature, categorizing these findings, summarizing and analyzing them, comparing them with our work, and of course, referencing them properly. Let's dive in!

Slide 2: Identifying and Categorizing Related Works

So, our first step is to figure out our key themes. Let's say we're researching 'Machine Learning in Healthcare'. Our themes could be 'Machine Learning', 'Predictive Analysis', and 'Healthcare Applications'. Next, we hit the books - or rather, the databases. Google Scholar, IEEE Xplore, and the ACM Digital Library are some great places to start. Once we have our articles, we need to sort them into categories. We might group them by how relevant they are to our work, what methods they used, their theoretical approach, or how they've been applied in the field.

Slide 3: Analyzing and Comparing Related Works

Once we've got our articles sorted, it's time to get to the meat of it. For each category, we summarize the relevant findings and methods from the articles, and then we look at how these findings relate to our own research. Do they back up our argument? Maybe they contradict it, or give a completely different perspective. This is also the time to highlight the similarities and differences between our work and the related works. We could be looking at categories like Theoretical Approaches, Methodologies, Applications, Experimental Results, or Reviews and Surveys.

Slide 4: Conclusion

But remember, the Related Works section isn't just a summary of what's been done before. It's our chance to show how our work builds on what's come before, or maybe even how it takes a completely new direction. It helps us highlight the significance and novelty of our research. And let's not forget, citing our sources correctly is absolutely crucial. We're standing on the shoulders of giants, and we need to give them their due.

08

Slide 1 - Introduction:

Good afternoon, everyone! Today, we're going to delve into the crucial aspect of describing the importance of a research topic in the field of computer science. It's essential to engage readers and establish the relevance of our research. So, let's dive in!

Slide 2 - Types of Importance (Practical and Theoretical):

Let's explore two key types of importance that we can emphasize in our research introductions. First, we have practical importance. This is all about highlighting real-world applications and the potential benefits our research can offer. Imagine developing an efficient algorithm for real-time object detection in autonomous vehicles. Such advancements can significantly enhance the safety and reliability of self-driving cars. Moving on to theoretical importance. Here, we're focused on advancing the theoretical foundations of computer science. Our research can challenge existing theories or provide new insights. For example, imagine investigating the complexity class of a newly proposed problem. By doing so, we gain a deeper understanding of computational limits and broaden our knowledge of what's feasible.

Slide 3 - Types of Importance (Societal and Technological):

Now, let's explore two more categories of importance. First up, we have societal importance. Our research has broader societal impacts, addressing challenges or improving quality of life. Consider developing a secure and privacy-preserving framework for handling healthcare data. This has societal importance as it enables confidential sharing of medical information, leading to more accurate diagnoses and improved patient care. Next, we have technological importance. This is about advancing existing technologies or enabling new advancements. Imagine developing a more efficient routing protocol for wireless sensor networks. This advancement can extend network lifetime, reduce energy consumption, and enhance the scalability and reliability of IoT-based systems.

Slide 4 - Conclusion:

To wrap things up, we've explored the significance of describing the importance of our research topics. Remember, supporting evidence and examples are crucial in conveying this importance effectively. By engaging readers and establishing relevance, we can highlight the practical, theoretical, societal, and technological impacts of our work. So, let's put emphasis on the significance of our research and make a lasting impact on the field of computer science.

09

Slide 1 - Introduction:

Hey there, today we'll be talking about the importance of describing novelty in research articles in the field of computer science. Describing novelty is a way to highlight the significance and originality of the research you're conducting. So, what exactly is novelty? Novelty refers to the extent to which the research presents new insights or advances existing knowledge in the field.

Slide 2 - Categories of novelty claims:

Now, let's take a closer look at the different categories of novelty claims. There are four main categories that researchers can make in the introduction of their articles: conceptual, methodological, empirical, and theoretical novelty.

* Conceptual novelty refers to the introduction of new ideas or concepts to the field. For example, a new algorithm or a new approach to solve a problem.
* Methodological novelty refers to the introduction of a new methodology or technique to the field. For example, a new experimental design or a new simulation model.
* Empirical novelty refers to the introduction of new empirical evidence or data to the field. For example, presenting new findings from a dataset or a new case study.
* Theoretical novelty refers to the introduction of a new theoretical framework or perspective to the field. For example, proposing a new theory or a new hypothesis.

Slide 3 - Examples of novelty claims in computer science:

Let's look at some examples of novelty claims in computer science to get a better understanding of these categories.

* Conceptual novelty: A machine learning algorithm that combines deep neural networks with graph theory to improve the accuracy of predicting protein-protein interactions.
* Methodological novelty: A simulation-based optimization approach that integrates agent-based modeling and evolutionary algorithms to improve the efficiency of supply chain management.
* Empirical novelty: Novel empirical evidence on the impact of gamification on student engagement in online learning environments.
* Theoretical novelty: A novel theoretical framework for understanding the relationship between social media use and mental health outcomes among adolescents.

Slide 4 - Conclusion:

In conclusion, it's important to justify your novelty claims with evidence and acknowledge existing knowledge in the field. When writing the introduction of your article, consider which category of novelty your research falls into and clearly explain how it extends or challenges existing knowledge.

10

Slide 1: Introduction

Hello everyone, today we're going to talk about writing an effective overview for a research article in computer science. This is an essential skill, as a good overview helps your readers navigate your work. It's like giving them a roadmap of your research. A well-written overview can help highlight the importance of your research and bring out the novelty of your findings. So, let's dive in.

Slide 2: Steps to Write an Overview (Part 1)

Writing an overview involves a few crucial steps. Firstly, you want to mention the main topics that you're going to discuss in your article. Keep it brief, but make sure these topics follow the same order as they appear in the article. It's like giving your readers a sneak peek into the chapters of a book.

Secondly, you need to describe your methodology. Again, keep this short. You're not going into the nitty-gritty details here. You just want to give your readers an idea of the approach you've taken to tackle the problem.

Slide 3: Steps to Write an Overview (Part 2)

Moving on to the third step, you want to give your readers a glimpse of the results or findings. But remember, this isn't the place for detailed explanations or analysis. You're just whetting the readers' appetite, making them curious to learn more.

The fourth step is to mention the structure of the paper. This is like providing a map to your readers. It helps them navigate through your paper and understand how your argument progresses. But remember, the key here is brevity.

Slide 4: Conclusion

In conclusion, writing an effective overview isn't too hard. It's about being clear, concise, and giving your readers a glimpse into what's coming. It's about generating interest and making it easy for your readers to understand your research. Remember, your overview is like a roadmap for your paper.

11

Slide 1: Introduction

Today we'll be discussing how to effectively refer to figures, like graphs, diagrams, and screenshots, in research articles. As computer science students, you'll often need to include these visuals in your work to illustrate and support your ideas.

Slide 2: Direct Reference & Sentence Incorporation

Let's start with the first two methods for referring to figures. The Direct Reference method involves simply mentioning the figure number within your text, typically in parentheses. It's great when the figure provides supplementary information. Imagine saying, 'We use a machine learning model to detect fraudulent transactions (Figure 1).

The second method is Sentence Incorporation. This is when you weave the figure reference into your sentence itself. It works best when the figure is central to the discussion. For example, 'As shown in Figure 2, our neural network architecture consists of multiple layers, including input, hidden, and output layers.

Slide 3: Descriptive Reference & Citation-Style Reference

Moving on, our third method is the Descriptive Reference. Use this when the figure contains complex information that needs a brief explanation. An instance would be, 'Figure 3 presents a flowchart outlining the steps of our proposed algorithm, from data preprocessing to the final output.'

Finally, there's the Citation-Style Reference. Treat the figure like a citation, so you can refer to it without disrupting the flow of your sentence. An example could be, 'Our findings are consistent with previous research, which indicates a rise in the use of quantum computing in recent years (see Figure 4).

Slide 4: Conclusion

To wrap up, we've explored four methods of referring to figures: Direct Reference, Sentence Incorporation, Descriptive Reference, and Citation-Style Reference. Remember, when you include figures, ensure they're numbered, captioned descriptively, and clearly legible. As you embark on writing your research articles, these techniques will be invaluable. Happy writing, folks!

12

Slide 1

Hello everyone, today's focus is on how to effectively refer to tables in your research articles. As computer science students, tables are often a fundamental part of presenting and comparing data in your work.

Slide 2

We have four main methods to refer to tables in your writing. The first one is Direct Reference. It's pretty simple - you just mention the table number in your text, often in parentheses. It's particularly useful when the table provides additional data. An example might be, 'The performance comparison between different machine learning algorithms is presented (Table 1).

The second method is Sentence Incorporation. Here, you weave the table reference directly into your sentence. This is perfect when the table is a crucial part of the discussion. For example, 'Table 2 shows the distribution of data points among different classes in the dataset.'

Slide 3

Next up, we have Descriptive Reference. Use this when the table contains complex information that needs a brief explanation. For instance, 'As summarized in Table 3, the processing times of the proposed algorithm are significantly lower than those of existing methods.

Finally, there's the Citation-Style Reference. Treat the table like a citation, so you can refer to it without disrupting the flow of your sentence. An example could be, 'Our results, showing an increase in computational efficiency across multiple tasks (see Table 4), are consistent with the hypothesis that our new algorithm performs better than existing methods.

Slide 4

To wrap up, we've covered four methods of referring to tables: Direct Reference, Sentence Incorporation, Descriptive Reference, and Citation-Style Reference. When you include tables in your articles, remember to number them, give them descriptive titles, and ensure they're clearly legible.

13

Slide 1

Hello everyone. Today we're going to talk about an important aspect of writing research papers, specifically how to reference equations. Why is this important? Well, it helps guide your readers through your logical and mathematical reasoning. There are different styles to do this and today, we're going to explore these methods. By the end of this presentation, you'll understand different ways to refer to equations in a computer science research article, helping you make your arguments clearer and more convincing. Let's dive in.

Slide 2

Alright, let's start with the first two methods: In-Line References and Parenthetical References. In-Line References are casual and direct. We use them when the equation is really relevant to what we're saying in that sentence. For example, we might say, 'From equation 1, we can see that the time complexity of our algorithm is O(n log n).' Simple, right? On the other hand, Parenthetical References are a bit more formal. We use these when our equation supports what we're saying, but isn't the main focus. So, we might say something like, 'The time complexity of our algorithm is O(n log n),' and then, in parentheses, 'see equation 1.' We're basically telling our readers, 'Hey, if you want to see how we got here, check out equation 1.

Slide 3

Okay, moving on to the next two methods: Direct References and Narrative References.

With Direct References, we're straight to the point. We just refer to the equation by its number without even saying the word 'equation'. We use this style when we have lots of equations, and it's clear from the context what we're referring to. We might say, 'As shown by 1, the time complexity of our algorithm is O(n log n).' It's like we're saying, 'Hey, remember equation 1? That's why our time complexity is O(n log n).' Narrative References, though, are a bit more descriptive. We use these when we want to describe the nature or the significance of the equation as part of our text. So, we could say, 'Our conclusion about the time complexity of our algorithm being O(n log n) is drawn from our main result, which is equation 1.' We're giving equation 1 a bit more of a spotlight here, aren't we?

Slide 4: Conclusion

Before we finish, let's recap a few key points. Firstly, it's really important to be consistent with your referencing style throughout your paper. And remember, always use a clear numbering system for your equations. It helps your reader to follow your argument more easily.

14

[Slide 1]

Hello everyone! Today, we'll be talking about prepositions, conjunctions, and transitions in the English language. These elements are essential for connecting ideas and providing clarity in our sentences. We'll be focusing on examples related to computer science, which will be helpful for all of you as you study this fascinating subject.

[Slide 2]

First, let's discuss prepositions. Prepositions help us connect noun phrases in a sentence and show the relationship between a noun or pronoun and other words. Some common prepositions are 'in', 'on', and 'with'. Let's look at a few examples from computer science:

* The data is stored *in* the database.
* She is working *on* the software.

The algorithm runs faster *with* optimization.

* In these examples, the prepositions help us understand the relationships between different parts of the sentence.

[Slide 3]

Now, let's move on to conjunctions and transitions. Conjunctions connect clauses, or groups of words with a subject and verb, in a sentence. Some common conjunctions are 'and', 'but', 'or', 'so', and 'because'. Here are some examples:

* The code is efficient *and* easy to read.
* The program crashed *because* there was a bug.

You can use Python *or* JavaScript for this project.

* In these sentences, conjunctions link clauses together to create more complex ideas.

Transitions, on the other hand, connect sentences and show the relationship between them. They make the text flow smoothly and can indicate addition, contrast, cause and effect, and more. Let's check out a couple of examples:

* The programmer fixed the bug. *However*, some performance issues still remained.
* The team implemented a new feature. *As a result*, the software became more versatile.

In these cases, transitions help us understand how ideas in different sentences are related.

[Slide 4]

To wrap up, understanding prepositions, conjunctions, and transitions is crucial for clear communication. Using them properly helps our sentences flow better and connects our ideas more effectively. This knowledge is especially important in technical fields like computer science, where conveying complex ideas is often necessary.

Thank you for your attention! I hope you find this information useful as you continue your studies in computer science and improve your English language skills.

15

Slide 1: Introduction

Hello everyone, today we're going to talk about noun phrases in the context of computer science. We'll explore how these phrases can enhance our understanding and communication when discussing computer science topics. Noun phrases consist of nouns, prepositions, and additional details that help to provide more information.

Slide 2: Reasons to Add Noun Phrases

The first reason to add noun phrases is to give more information. Let's look at an example:

Without a noun phrase: "The developer made a new solution."

With a noun phrase: "The developer made a new solution with Python, a helpful language."

As you can see, the sentence with the noun phrase provides extra details that make it more informative.

Another reason to use noun phrases is to make sentences clearer. Here's an example:

Without a noun phrase: "The group worked together on the project."

With a noun phrase: "The group worked together on the project about machine learning."

The noun phrase helps us understand the focus of the project, making the sentence clearer.

Slide 3: Reasons to Add Noun Phrases (Continued)

The third reason to use noun phrases is to create variety in our speech and writing. Let's see an example:

Without a noun phrase: "The data person found important information."

With a noun phrase: "The data person found important information from the big dataset with many data points."

The sentence with the noun phrase is more interesting and less repetitive. So, using noun phrases can improve our writing and speech in computer science.

Slide 4: Conclusion

To sum up, noun phrases are valuable for enhancing communication in computer science. They help to provide extra details, improve clarity, and create variety in our speech and writing. By applying noun phrases in the computer science context, we can better understand and communicate complex concepts, making our discussions more engaging and informative. Thank you for listening!

16

Slide 1: Introduction

Hello everyone! Today, we'll discuss the importance of transition words and phrases in academic writing. These words and phrases make our writing coherent and smooth, guiding readers through the text.

Slide 2: Categories of Transition Words and Phrases

We'll cover five main categories of transition words and phrases. Addition words like "also" and "moreover" add information. Contrast words, such as "however" and "but," show differences between ideas. Cause and effect words like "therefore" and "consequently" show reasons or results.

Time-related words, including "meanwhile" and "then," help us show when something occurs or the order of events. Sequence words like "first" and "next" help us arrange our ideas in order.

Slide 3: Examples in Computer Science Context

Let's look at two examples from computer science. The first one is: "Python is popular for beginners. However, some prefer JavaScript for web development." Here, "however" is a contrast word.

In the second example, we have two paragraphs:

Paragraph 1:

"Python is known for simplicity and readability. Moreover, it has many libraries and frameworks."

Paragraph 2:

"On the other hand, JavaScript is primarily used for client-side scripting in web development."

Here, "on the other hand" is a contrast word that connects ideas across paragraphs.

Slide 4: Conclusion

To sum up, using transition words and phrases effectively improves our writing. They help create clear, coherent, and well-structured texts, making it easier for readers to understand and follow our ideas. Remember to use these words and phrases to enhance your writing. Thanks for listening!

17

[Slide 1: Introduction]

Hello everyone, today we're going to talk about clauses and conjunctions in the English language. Clauses are groups of words that contain a subject and a verb. Conjunctions are the connectors that join clauses together, helping us create more complex sentences and convey relationships between ideas. Understanding clauses and conjunctions is important for improving your English skills.

[Slide 2: Coordinating Conjunctions]

Now, let's discuss the first type of conjunctions, called coordinating conjunctions. These conjunctions connect independent clauses, or main clauses, with equal importance. You can remember the common coordinating conjunctions using the acronym FANBOYS. It stands for "For," "And," "Nor," "But," "Or," "Yet," and "So."

Here's an example: "She wanted to go to the store, but it was closed." The conjunction "but" connects two independent clauses and emphasizes the contrast between her wanting to go to the store and the store being closed.

[Slide 3: Subordinating Conjunctions]

The second type of conjunctions is subordinating conjunctions. These conjunctions connect a dependent clause, which cannot stand alone, to an independent clause, which can stand alone. Examples of subordinating conjunctions include "Because," "Since," "Although," "If," "When," "While," and "Before."

Let's look at an example: "We will go for a picnic if the weather is nice." The conjunction "if" links the dependent clause "if the weather is nice" to the independent clause "We will go for a picnic." It shows a condition that must be met for the main action to occur.

[Slide 4: Conclusion]

To wrap up, conjunctions play a crucial role in creating clear, coherent, and well-structured sentences by connecting clauses in the English language. By understanding and using coordinating and subordinating conjunctions, you'll be able to express your ideas more effectively in English. So, keep practicing, and you'll see improvements in your language skills. Thank you for listening!

18

[Slide 1]

Hello everyone! Today, we're going to talk about the passive voice in English and its importance in computer science academic writing. This is especially useful for those of you studying computer science and looking to improve your English writing skills.

[Slide 2]

Now, let's discuss how and when to use the passive voice. In computer science writing, the passive voice is often used to focus on the process, results, or research rather than who did the work. This makes the writing more objective. Here are some examples:

* For a process: "The software was updated to fix the bug."
* For results: "The test accuracy was improved by 10%."
* For research: "The model was trained using a large dataset."

[Slide 3]

However, it's essential to know that not all verbs can be used in the passive voice. Intransitive verbs, which are verbs that don't have an object, cannot be made passive. This is because there's no object to move to the subject position in the sentence. Here are some examples of intransitive verbs: sleep, arrive, and laugh. Remember, only transitive verbs, those with an object, can be used in the passive voice.

[Slide 4]

In conclusion, the passive voice is an important tool for writing clear and objective sentences in computer science academic writing. It allows you to emphasize the action, not the doer, and helps keep the focus on the ideas. However, keep in mind that you should use the passive voice only with transitive verbs. Good luck with your writing, and I hope this information helps you create better academic papers!

19

Slide 1: Introduction

Hello everyone, today we'll discuss the importance of using appropriate tenses in computer science writing. We'll focus on three main tenses: Present Simple, Present Perfect Simple, and Past Simple.

Slide 2: Present Simple & Present Perfect Simple

First, let's talk about the Present Simple tense. We use this tense to describe things that are true now and usually remain true. It helps us understand basic ideas and current developments in computer science. For example, we can say, "A neural network consists of layers of interconnected nodes."

Next is the Present Perfect Simple tense. We use this tense to talk about events that started in the past but are still important now. This is often used to show the significance of previous research in an introduction. An example of this tense is, "Recent research has demonstrated that machine learning tools can tackle difficult problems."

Slide 3: Past Simple

Now, let's move on to the Past Simple tense. We use this tense to describe actions that were completed in the past, usually related to a project or an experiment. For instance, we can say, "The team developed the tool and tested its effectiveness using a large dataset."

Slide 4: Conclusion

In conclusion, using appropriate tenses is crucial for writing clear and easy-to-follow computer science research papers. Remember to use the Present Simple tense for fundamental concepts, the Present Perfect Simple tense for ongoing relevance of past research, and the Past Simple tense for project execution and experimentation. Thank you for your attention, and I hope this information helps you in your future computer science writing endeavors.

20

Slide 1: Introduction

* Hi everyone! Today, we'll discuss the present simple tense and its role in computer science academic writing. We'll cover its use in explaining basic concepts, describing current states, and discussing theories.

Slide 2: Main Details (Part 1)

* + So, let's start with explaining general facts. We use the present simple tense to share basic ideas and theories in computer science.
    - For example, "Machine learning models learn from data."
  + We also use this tense to describe the current state, properties, or features of a technology or system.
    - Like when we say, "A neural network has interconnected nodes."

Slide 3: Main Details (Part 2)

* + Moving on, we use the present simple tense to discuss widely accepted theories and principles in computer science. This helps us understand the foundation of the subject.
    - An example is, "Information travels using the Internet Protocol."
  + Using the present simple tense in this way is important for conveying clear and easily understandable information.

Slide 4: Conclusion

* In conclusion, the present simple tense is essential for providing clarity in computer science academic writing. It helps readers understand general facts, current states, and foundational theories. Remember to use the present simple tense effectively in your writing to make it accessible and engaging. Thank you for listening!

21

Slide 1: Introduction

* Hi everyone! Today, we'll discuss the present perfect simple tense and its role in computer science academic writing. We'll explore its use in emphasizing the ongoing importance of past developments, achievements, and discoveries.

Slide 2: Main Details (Part 1)

* First, let's talk about how the present perfect simple tense helps highlight the ongoing importance of past developments.
  + For example, "Researchers have made significant advancements in artificial intelligence."
* We also use this tense to describe actions that occurred in the past but have a direct impact on the present situation.
  + Like when we say, "Computer scientists have improved data storage techniques, enabling us to store more information."

Slide 3: Main Details (Part 2)

* Next, let's look at how we use the present perfect simple tense to describe past events when the exact time is not important, but the event's impact on the present is relevant.
  + An example is, "Scientists have discovered new methods for optimizing algorithms."
* This tense is particularly useful in the Introduction sections of academic writing as it helps convey the connection between past events and their present significance.

Slide 4: Conclusion

* In conclusion, the present perfect simple tense is essential for demonstrating the ongoing relevance of past developments in computer science academic writing. It effectively connects past actions with present results and helps us communicate the importance of past developments. Remember to use the present perfect simple tense effectively in your writing to make it engaging and informative. Thank you for listening!

22

Slide 1: Introduction

* Hello everyone! Today, we're going to talk about the past simple tense and its importance in computer science academic writing, especially in the Methods section.

Slide 2: Reasons & Situations (Part 1)

* First, let's discuss why we use the past simple tense. We use it to describe past actions or procedures, like when we say, "We collected data and preprocessed it." This helps the reader understand how the research was done.
* We also use the past simple tense to mention the tools or software we used in our work. For example, "The researchers used Python to create a machine learning model."

Slide 3: Reasons & Situations (Part 2)

* Another reason we use the past simple tense is to explain the experimental setup. For instance, we might say, "Participants were split into two groups, and each group had a different task."
* Lastly, we use the past simple tense to talk about how we analyzed the data and evaluated the outcomes. Like when we say, "We used statistical methods to check the importance of the results."

Slide 4: Conclusion

* In conclusion, the past simple tense is crucial for clearly describing the methodology in computer science academic writing. It helps readers understand the steps we took, the tools we used, and how we analyzed the results. Remember to use the past simple tense effectively in your writing to make it easy for your audience to follow. Thank you for listening!

23

Slide 1: Introduction

* Hello everyone! Today, we will talk about hedging in academic writing, especially in the field of computer science. We'll discuss why it's important and how it helps create respectful conversations among researchers.

Slide 2: Hedging Words

* Hedging means using careful words to make your ideas less strong or direct. Some examples of these words are might, could, may, possibly, seems, and generally. We use them to show that we are open to other ideas and to avoid making statements that are too strong or certain.

Slide 3: Computer Science Context

* Computer science is a complex field where different ways can solve problems, and new information can change our understanding. That's why using hedging words is important. They encourage thoughtful conversation among researchers and show that we respect other opinions.

Slide 4: Conclusion

* In summary, hedging helps us share our findings accurately and carefully. It makes our communication in computer science more respectful and supports the overall quality of academic writing. Remember to use hedging words when writing about your research to make your ideas clear and open to other perspectives. Thank you for listening!

24

Slide 1: Introduction

* Hi everyone! Today we're going to talk about boosters in academic writing and their importance in computer science. We'll learn how to balance certainty and credibility when using these words.

Slide 2: Boosters in Context

* Boosters are words that make sentences stronger and more certain, like definitely, clearly, undoubtedly, and always. We use them in computer science writing to express confidence and emphasize important points. These words show that we believe our results are accurate and can be trusted.

Slide 3: Cautious Use of Boosters

* It's important to use boosters only when the evidence strongly supports our idea. If we use too many boosters, our writing might seem overly confident, which isn't good. So, we should aim for balanced and well-supported arguments in our writing.

Slide 4: Conclusion

* To wrap up, boosters help us emphasize important points in our writing. When writing about computer science, we should use them cautiously and appropriately. By balancing the use of boosters, we can maintain credibility and trustworthiness in our work. Thanks for listening!

25

Slide 1: Introduction

* Hello everyone! Today, we'll discuss approximation in English and its importance in computer science academic writing. We'll learn how using approximation can make our writing clearer and simpler.

Slide 2: Approximation Words

* Approximation words, like about, around, nearly, and almost, help us indicate numbers or sizes without being exact. They allow us to convey general ideas and avoid being too specific when it's not necessary.

Slide 3: Computer Science Context

* In computer science writing, we use approximation when precise numbers aren't crucial. It helps simplify information, making it easier to understand. For example, we might say "The software takes around 10 minutes to install," or "The system can handle nearly 200 users at once."

Slide 4: Conclusion

* To sum up, using approximation in computer science writing makes our work more accessible. By balancing details and simplicity, we can create clearer and more readable texts. Remember to use approximation words when needed to enhance your writing's clarity and readability. Thanks for listening!