01

Abstract

An abstract for a short research article summarizes the paper's content concisely and highlights its key points. It should be no more than 200 words, and generally includes the following components:

1. Introduction (Move 1): Introduce the topic and its importance briefly. Example: "Recent advances in machine learning have led to a surge in deep learning algorithms, but training these models can be computationally expensive."
2. Objective/Research Question (Move 2): State the main objective of your study. Example: "This study investigates techniques to reduce the computational costs of training deep learning models without sacrificing accuracy."
3. Methodology (Move 3): Briefly describe the methods used. Example: "We implemented and compared three pruning algorithms on a popular deep learning architecture, ResNet-50, and evaluated their performance on the CIFAR-10 dataset."
4. Results (Move 4): Summarize the main findings of your research. Example: "Our results show that algorithm A achieved up to 40% reduction in computational complexity, while maintaining comparable accuracy to the original model."
5. Conclusion/Implications (Move 5): Conclude by stating the significance of the findings and their potential impact. Example: "The proposed pruning techniques can significantly reduce computational resources required for training deep learning models, enabling wider adoption in resource-constrained environments."

Remember to keep the language clear and concise, avoid jargon, and maintain a logical flow throughout the abstract

02

Introduction section

To write an effective introduction for a short computer science research article, you should use several rhetorical moves to establish context, highlight the significance of your study, and present your research goals. Here are some common rhetorical moves and examples for each:

1. Introduce the research topic and context (Move 1): Start by briefly presenting the general area of research, setting the stage for your specific focus. Example: "Deep learning has revolutionized various fields, including image recognition, natural language processing, and reinforcement learning."
2. Identify the research gap (Move 2): Point out limitations or areas that have not been well-addressed in existing research. This helps justify your study. Example: "However, deep learning models often require substantial computational resources, limiting their applicability in resource-constrained environments."
3. State the significance and relevance of the research (Move 3): Explain why addressing the identified gap is important and how it could contribute to the field. Example: "Reducing the computational cost of training deep learning models could expand their adoption and enable their use in real-time applications, such as robotics and mobile devices."
4. Present the research objective or question (Move 4): Clearly state the main goal or research question your study aims to address. Example: "This study investigates efficient techniques to reduce the computational costs of training deep learning models without compromising their accuracy."
5. Provide an outline of the article (Move 5, optional): Give a brief overview of the organization of the paper, helping guide the reader through the subsequent sections. Example: "We first discuss related work on model pruning, followed by the methodology, experiments conducted, and finally the results and their implications."

When writing an introduction, use clear and concise language, avoid excessive jargon, and ensure a logical flow between the different rhetorical moves.

03

Method section

The method section of a research article in computer science provides a detailed description of the methodology and procedures used to conduct the study. It should enable other researchers to replicate the experiment or study and assess the validity of the results. Here are some typical rhetorical moves and a short computer science example for each category:

1. Research Design:

Clearly state the research design or approach used in the study.

Example: "This study adopts an experimental research design to evaluate the performance of the proposed algorithm in comparison to existing methods."

2. Participants or Data Collection:

Describe the participants involved in the study or the data collection process.

Example: "A total of 100 undergraduate students from various computer science programs were recruited as participants for the user study."

3. Variables and Measures:

Specify the variables studied and the measures used to collect data.

Example: "The independent variable in this study is the level of encryption (low, medium, high), and the dependent variable is the time taken to encrypt a file measured in milliseconds."

4. Experimental Setup or Software Environment:

Explain the experimental setup, hardware, software, or simulation environment used.

Example: "The experiments were conducted on a server with an Intel Core i7 processor, 16GB RAM, and Ubuntu 20.04 operating system. The programming language used was Python 3.9."

5. Data Collection Procedure:

Describe how the data was collected or experiments were conducted.

Example: "Participants were individually seated in front of a computer and provided with a set of instructions. They were then asked to perform a series of tasks using the proposed system while their interactions were recorded."

6. Data Analysis:

Outline the statistical or analytical methods employed to analyze the data.

Example: "Descriptive statistics, such as means and standard deviations, were calculated for each variable. A one-way analysis of variance (ANOVA) was performed to compare the mean performance across different encryption levels."

7. Ethical Considerations:

Address any ethical considerations, such as informed consent or data privacy.

Example: "Ethical approval was obtained from the Institutional Review Board, and all participants provided written informed consent before participating in the study. Participant data was anonymized and stored securely."

04

Results section

Here are some steps to write a Results section for a Computer Science research paper:

1. State the Main Findings:

Start by outlining the key findings. This is where you'll provide the answers to the research question(s) you've set. For example, "Our developed algorithm successfully improved the efficiency of data sorting by 25%."

2. Provide Details:

Next, provide more detailed findings. You should provide context to your results, with specifics and accurate details. For example, "The algorithm's efficiency was tested by comparing its sorting time against that of the standard algorithm, using datasets of varying sizes. The developed algorithm showed a consistent 25% decrease in sorting time across all datasets."

3. Use Visual Aids:

In many cases, results are best presented visually. Graphs, charts, tables, and diagrams can illustrate the results in a more digestible format. For example, "As shown in Figure 1, the developed algorithm's sorting time (represented by the red line) was consistently lower than that of the standard algorithm (represented by the blue line) across all dataset sizes."

4. Describe Negative Results:

All results should be reported, even if they were not as expected or if they contradict your hypothesis. This is important for maintaining scientific integrity. For example, "While the developed algorithm showed improved efficiency in sorting, it consumed 15% more memory compared to the standard algorithm, as shown in Table 2."

5. Summarize the Findings:

End the Results section with a summary of the findings. This should be a brief recap of the main points. For example, "In summary, our developed algorithm improved the efficiency of data sorting by 25%, albeit at the cost of increased memory usage."

6. Do Not Interpret:

Remember, the Results section is not where you interpret your findings. That is the job of the Discussion section. Keep your opinions, implications, and interpretations out of the Results section.

Here are some rhetorical moves to help present your results:

* Overview of Results: Provide a summary of the most significant findings. This gives the reader an initial understanding of the results.
* Sequential Presentation: Present your results in a logical sequence that aligns with the methods you used to gather and analyze the data. This makes it easier for the reader to follow and understand your findings.
* Highlighting Important Results: Highlight any results that are particularly important to your research question. This helps draw the reader's attention to key findings.
* Explanation of Figures and Tables: Clearly explain the content of any figures, tables, or other visual aids you use. This ensures that the reader fully understands the data presented.

Lastly, remember to be clear and concise in your writing. Use plain language and avoid unnecessary jargon. The goal is to convey your findings in a way that is easy for the reader to understand.

05

Discussion section

The discussion section of a research article is where you interpret your results, highlight significant findings, tie them to previous research, and discuss the implications of your research. Your goal is to explain what your results mean and why they are important. Here's a step-by-step guide:

1. Summarize the Main Findings

Begin by restating your research question and summarizing your primary findings. This should be a concise restatement of the results section, rather than a detailed reiteration.

Rhetorical Move: Restatement of Results

Example: Our analysis indicates a substantial improvement in the speed of sorting algorithms when using our proposed method, with a 30% decrease in time complexity compared to traditional quicksort.

2. Interpret the Findings

Discuss the significance of your results in relation to the hypotheses or questions posed in the introduction. What do your results mean in the context of your research?

Rhetorical Move: Interpretation of Results

Example: The reduction in time complexity signifies that our sorting method can handle larger data sets more efficiently, making it a potential candidate for big data applications.

3. Compare with Previous Research

Compare your results with previous studies. If your results align with prior research, explain how your study supports that work. If your results differ, provide possible reasons for the discrepancy.

Rhetorical Move: Comparison to Prior Work

Example: These results are in line with the findings of Doe et al. (2020), who suggested that modifying the pivot selection technique in quicksort could yield better performance. However, our method differs from theirs in that we implemented a hybrid approach...

4. Discuss Limitations

Acknowledge any limitations of your study. This might include issues with the methodology, data collection, analysis, or interpretation. This section should not be used to excuse poor research or analysis, but to provide an honest assessment of what your study did not cover.

Rhetorical Move: Statement of Limitations

Example: The primary limitation of our study was that we tested our method solely on synthetic datasets. Its effectiveness on real-world datasets remains to be seen...

5. Implications and Future Work

End by discussing the practical implications of your findings, and how they can be used in future research. Suggest areas for future study that could help extend or apply your research.

Rhetorical Move: Implications and Future Work

Example: Given the promising results of our study, there are several possible directions for future research. We could evaluate our sorting algorithm's performance on real-world datasets, explore different hybrid approaches...

Remember, clarity and concise writing is key in the discussion section. Avoid overly technical jargon unless necessary and be sure to explain all your points clearly.

06

References section

Whether you're using LaTeX or MS Word, the format of your references section largely depends on the citation style mandated by your target journal or conference. The most common styles in computer science include IEEE, ACM, and APA, among others.

Here's a general guide on how to prepare your references section:

In LaTeX

1. Using .bib file

This is a popular method because it automates the process and ensures consistent formatting. First, you need to create a .bib file. This is a bibliography database where each reference is stored with a unique identifier. Here is an example of how to format entries in the .bib file:

@article{key,

author = "Author Name",

title = "Title of the Article",

journal = "Journal Name",

year = "Publication Year",

volume = "Volume Number",

number = "Issue Number",

pages = "Page Numbers",

}

After creating the .bib file, place it in the same folder as your main .tex file. In your main .tex file, use the `\cite{key}` command to cite references in the text where `key` is the unique identifier of the reference in the .bib file.At the end of your document (where you want the references to appear), use the following commands:

\bibliographystyle{stylename}

\bibliography{bibfilename}

Here `stylename` is the name of the bibliography style (e.g., IEEEtran for IEEE format) and `bibfilename` is the name of your .bib file without the extension.

2. Using \bibitem command

This is a manual method where you directly write each reference in the LaTeX document. After the text of your paper, insert the following:

\begin{thebibliography}{99}

\bibitem{key}

Author Name, "Title of the Article," Journal Name, vol. Volume Number, no. Issue Number, pp. Page Numbers, Year.

\end{thebibliography}

In the text, use `\cite{key}` to cite the references where `key` is the unique identifier you provided in the `\bibitem`.

In MS Word

If you're using Word, you have two options:

1. Using a reference management tool

Word has a built-in tool for this, but there are also third-party tools like Mendeley, Zotero, and Endnote. These allow you to store references, automatically format them according to a selected style, and insert in-text citations. To use Word's built-in tool, navigate to the "References" tab and select the appropriate citation style. Click "Insert Citation" to add a new source. Fill in the details in the "Create Source" window. To create your bibliography, click "Bibliography" in the "References" tab and select "Works Cited". Word will automatically create a bibliography based on the sources you cited.

2. Manually

If you choose to write the references manually, ensure to strictly follow the chosen citation style. Here is a general example in IEEE format:

- Author Initials. Author Surname, "Title of article," in Journal Title, vol.#, no.#, pp. page number/s, Month Year.

For citing in the text, IEEE uses numbers in square brackets, e.g., [1], which refer to the corresponding numbered reference in the reference list.

Remember, the key to a good references section is consistency. Ensure every reference follows the same format, and all the citations.

07

Related works

Writing the Related Works section for a short research article in computer science is crucial because it provides context for your research. It shows how your work fits into the larger scientific dialogue and indicates your awareness of major themes and findings in your area of study. Here's a general step-by-step guide:

1. Identify Your Key Themes

Start by identifying the main themes or concepts of your research. For example, if your research is on "Machine Learning Algorithms for Predictive Analysis in Healthcare," key themes might include machine learning, predictive analysis, and healthcare applications.

2. Search for Relevant Literature

Conduct a literature search for articles that relate to your themes. Use databases like Google Scholar, IEEE Xplore, ACM Digital Library, etc.

3. Categorize Your Findings

Group the articles you've found into categories based on their relevance to your research. Categories can be based on methodologies used, theoretical approaches, applications, or any other significant factors.

4. Summarize and Analyze

For each category, summarize the relevant findings and methodologies from the articles. Also, analyze how these findings relate to your research. Are they supporting your argument? Contradicting? Providing a different perspective?

5. Compare and Contrast

Highlight the similarities and differences between your work and the related works. This can help to justify your research and identify its unique contribution.

6. Referencing

Be sure to properly cite all references using the citation style appropriate for your field (like APA, MLA, IEEE, etc).

Here are some categories of related works you may consider, along with examples specific to computer science:

1. Theoretical Approaches: This could include different theoretical approaches or algorithms used in the area of study. For example, if your research involves developing a new algorithm for sorting, you might discuss other sorting algorithms like quicksort, mergesort, heapsort, etc.

2. Methodologies: This includes the ways in which research has been conducted in your field. For instance, if your research is on artificial intelligence in game design, you could discuss other methods used to incorporate AI in games, such as rule-based systems, machine learning, or neural networks.

3. Applications: This refers to how research has been applied in the real world. For example, if your research is on data security in cloud computing, you might discuss how existing encryption or authentication techniques have been used in cloud environments.

4. Experimental Results: If your research involves an experiment or a new product, you can discuss how others have tested similar ideas or products. For instance, if your research is on performance optimization of database systems, you might discuss other benchmarking studies done in this area.

5. Reviews or Surveys: These are comprehensive summaries of the current state of research on a particular topic. For example, if your research is on the Internet of Things (IoT), you could cite a recent survey paper discussing the latest advancements, challenges, and future directions in IoT research.

Remember, the Related Works section should not only summarize existing literature but also show how your work builds upon or differs from these works, thereby establishing the significance and novelty of your research.

08

Importance

When describing the importance of a research topic in the introduction of a research article in the field of computer science, it is crucial to convey the significance and potential impact of the study. This helps to engage the readers and establish the relevance of the research. Here are some categories of different types of claims of importance, along with computer science examples for each category:

1. Practical importance: Highlight the real-world applications and potential benefits that the research can offer. Discuss how the findings can contribute to solving practical problems or improving existing systems.

Example: "The development of an efficient algorithm for real-time object detection in autonomous vehicles is of utmost importance as it can significantly enhance the safety and reliability of self-driving cars."

1. Theoretical importance: Emphasize how the research can advance the theoretical foundations of the field. Explain how it can expand knowledge, challenge existing theories, or provide new insights into fundamental concepts.

Example: "Investigating the complexity class of this newly proposed problem has theoretical importance, as it can lead to a deeper understanding of the inherent computational limits and provide insights into the boundaries of computational feasibility."

1. Societal importance: Discuss the broader societal impact of the research and how it can address societal challenges, improve quality of life, or contribute to social well-being.

Example: "The development of a secure and privacy-preserving framework for handling sensitive healthcare data has societal importance, as it can enable seamless and confidential sharing of medical information, leading to more accurate diagnoses and improved patient care."

1. Technological importance: Highlight how the research can advance existing technologies, enable new advancements, or improve efficiency and performance in practical applications.

Example: "The development of a more efficient routing protocol for wireless sensor networks is of significant importance as it can extend network lifetime, reduce energy consumption, and enhance the scalability and reliability of IoT-based systems."

Remember to provide supporting evidence, such as statistics, relevant studies, or real-world examples, to strengthen the claims of importance and create a persuasive case for the relevance and value of the research topic.

09

Novelty

Describing novelty in the introduction of a research article is an important aspect of academic writing, as it highlights the significance and originality of the research. Novelty refers to the extent to which the research presents new insights or advances existing knowledge in the field. There are various categories of novelty claims that researchers can make in the introduction of their articles. Here are some of them with examples from computer science:

1. Conceptual novelty: This refers to the introduction of new ideas or concepts to the field. For example, a researcher might introduce a new algorithm, a new model, or a new approach to solve a problem.

Example: "In this article, we introduce a novel machine learning algorithm that combines deep neural networks with graph theory to improve the accuracy of predicting protein-protein interactions."

1. Methodological novelty: This refers to the introduction of a new methodology or technique to the field. For example, a researcher might introduce a new experimental design, a new data analysis technique, or a new simulation model.

Example: "In this article, we propose a novel simulation-based optimization approach that integrates agent-based modeling and evolutionary algorithms to improve the efficiency of supply chain management."

1. Empirical novelty: This refers to the introduction of new empirical evidence or data to the field. For example, a researcher might present new findings from a dataset, a new case study, or a new experiment.

Example: "In this article, we present novel empirical evidence on the impact of gamification on student engagement in online learning environments."

1. Theoretical novelty: This refers to the introduction of a new theoretical framework or perspective to the field. For example, a researcher might propose a new theory, a new conceptual model, or a new hypothesis.

Example: "In this article, we present a novel theoretical framework for understanding the relationship between social media use and mental health outcomes among adolescents."

It is important to note that claims of novelty should be supported by evidence and clearly justified in the introduction of the article. Researchers should also acknowledge existing knowledge in the field and explain how their research extends or challenges it.

10

Overview

Writing an overview or a preview of the upcoming sections in the introduction for a short research article in computer science involves summarizing the major points that will be discussed in the body of the article. The overview will provide your readers with a roadmap of your research, which will help them to understand the flow of your argument, the importance of your research, and the novelty of your findings.

Here are the steps you can follow to write the overview:

* Briefly Mention the Main Topics: In the overview, briefly mention the main topics or points that will be discussed in the article. These should follow the order in which they appear in the article.
* Describe the Methodology: Briefly describe the method or approach you will take to address the problem.
* Preview the Results: Give a glimpse of the results or findings, but do not go into detail. This will create interest and encourage readers to continue reading.
* Mention the Structure: Briefly explain the structure of the paper. This helps the reader navigate through the content and understand the progression of your argument.

Here is a generic example for a computer science article:

This research article explores the potential of blockchain technology in bolstering data security in cloud computing environments. Section 1 provides an in-depth discussion on the current security challenges in cloud computing. Section 2 introduces blockchain technology, focusing on its inherent security features. In Section 3, a novel blockchain-based security framework for cloud data storage is proposed and its implementation steps are detailed. Section 4 presents a series of tests that evaluate the effectiveness of the proposed framework, with results indicating an enhanced level of data security. Finally, Section 5 discusses the broader implications of these findings and potential avenues for future research.

Remember, the objective of the overview is to guide your reader, making it easier for them to understand the content of your article. Be clear and concise, and avoid going into too much detail.

11

Referring to figures

When writing a research article, you often need to refer to figures, such as graphs, screenshots, and diagrams, to support your arguments or illustrate your points. Here are four different ways to refer to figures in a research article:

1. Direct Reference:

This is when you refer to the figure by its number in the text, usually in parentheses. This is a straightforward approach typically used when the figure provides supplementary information, rather than being a primary focus of the discussion.

Example: "We utilize a machine learning model for the detection of fraudulent transactions (Figure 1)."

2. Sentence Incorporation:

This approach integrates the figure reference directly into your sentence. It is generally used when the figure is a central part of the discussion, and you want to draw the reader's attention to it.

Example: "Figure 2 illustrates the architecture of the neural network, consisting of multiple layers such as the input layer, hidden layers, and the output layer."

3. Descriptive Reference:

This approach is used to highlight the content of the figure in your text. It is useful when the figure contains complex information that needs to be explained briefly in the text.

Example: "The flowchart presented in Figure 3 outlines the steps of the proposed algorithm, from data preprocessing to the final output."

4. Citation-style Reference:

In this method, you treat the figure as you would a citation. This style is often used when you want to refer to a figure without interrupting the flow of the sentence.

Example: "Our findings are in line with previous research, indicating a rise in the use of quantum computing in recent years (refer to Figure 4)."

Remember that all figures should be numbered in the order they appear in the text, have a brief but descriptive caption, and should be clearly legible. Effectively referring to figures should enhance your discussion rather than distract from it. Your paper should also be comprehensible even if the reader doesn't look at every figure you mention.

12

Referring to Tables

Referring to tables in a research article is similar to referring to figures. Here are several ways to refer to tables:

1. Direct Reference

In this approach, you refer to the table by its number in the text, often in parentheses. This method is commonly used when the table offers additional or supporting data to your text. For example, "The performance comparison between different machine learning algorithms is presented (Table 1)."

2. Sentence Incorporation

This involves integrating the table reference into your sentence. This is typically used when the table is an essential part of the discussion and you want to draw the reader's direct attention to it. For example, "Table 2 shows the distribution of data points among different classes in the dataset."

3. Descriptive Reference

This is used when you want to highlight the content of the table in your text. This approach is useful when the table contains complex or comprehensive information that needs to be briefly explained in the text. For example, "As summarized in Table 3, the processing times of the proposed algorithm are significantly lower than those of existing methods."

4. Citation-style Reference

In this method, you treat the table as you would a citation. It's typically used when you want to refer to a table without interrupting the flow of the sentence. For example, "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."

Remember that all tables should be numbered in the order they appear in the text and have a concise but informative title. As with figures, the key to effectively referring to tables is to do so in a way that enhances your discussion rather than distracts from it. The text should also make sense even if the reader does not look at every table you refer to.

13

Referring to Equations

Referring to equations in a research article is vital for guiding your readers through your logical and mathematical reasoning. The precise method you use can depend on the length of the paper, the complexity of the equations, and the overall structure of your argument. However, there are some general guidelines you can follow. Here are four common methods to refer to equations in a computer science research article:

1. In-Line Reference

This is a casual and direct way to refer to equations. You might use it in less formal settings or when the equation is directly relevant to the sentence at hand. An in-line reference would typically use the word "equation" and the equation's number within parentheses. For example, "From the equation (1), we can infer that the time complexity of the algorithm is O(n log n)."

2. Parenthetical Reference

This is similar to an in-line reference but placed at the end of a sentence or statement, and is a formal way of referencing. It is often used when the equation supports the sentence, but is not the primary focus of it. For example, "The time complexity of the algorithm is O(n log n) (see equation 1)."

3. Direct Reference

This is when you directly refer to the equation by its number without using the word "equation". This style is often used when the document contains many equations, and the context makes it clear that a number refers to an equation. For example, "As demonstrated by (1), the time complexity of the algorithm is O(n log n)."

4. Narrative Reference

This is a more descriptive way of referring to equations and is used when you want to describe the nature or significance of the equation as part of the text. For example, "Our conclusion about the time complexity of the algorithm being O(n log n) is drawn from our primary finding, equation (1)."

Remember, it's essential to maintain consistency in your referencing style throughout the document. It's also crucial to use an appropriate numbering system for your equations, typically sequential numbers, and to make sure these numbers are clearly displayed next to the relevant equations. This could be done either to the right of the equation in parentheses or above or below the equation, depending on the formatting style of the article.

14

Linking expressions

Prepositions, conjunctions, and transitions help connect ideas and make sentences clear.

1. Prepositions connect noun phrases and show the relationship between a noun or pronoun and other words in a sentence. They often indicate location, direction, time, or other relationships. For example:

* The data is stored *in* the database.
* She is working *on* the software.
* The algorithm runs faster *with* optimization.

In these examples, the prepositions 'in', 'on', and 'with' connect the noun phrases ('the database', 'the software', and 'optimization') to the rest of the sentence.

1. Conjunctions are words that join two or more clauses (groups of words with a subject and verb). Common conjunctions include 'and', 'but', 'or', 'so', and 'because'. For example:

* 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.

Here, the conjunctions 'and', 'because', and 'or' connect the clauses in each sentence.

1. Transitions connect sentences by showing the relationship between sentences or paragraphs, making the text flow smoothly. They can indicate addition, contrast, cause and effect, or other connections.

Example 1:

The programmer fixed the bug in the code. *However*, some performance issues still remained.

In this example, the transition 'However' connects the two sentences by showing a contrast between the bug being fixed and the remaining performance issues

Example 2:

The team implemented a new feature. *As a result*, the software became more versatile.

In this case, the transition 'As a result' connects the sentences by showing the cause and effect relationship between implementing a new feature and the software becoming more versatile.

15

Prepositions

Noun phrases are groups of words with a noun as the main word.

For example: "the new language"

Prepositions show the relationship between the noun or pronoun and other words in the sentence. Common prepositions include "in," "on," "at," "with," "by," "for," "before," and "after."

You can add a noun phrase to a sentence to give more details about time, place, manner, or other relationships. For example:

Original: He found the bug. It took two hours.

With noun phrase: He found the bug after two hours.

Reasons to add a noun phrase:

a. Give more information: Noun phrases give extra details about languages, tools, or processes.

Example: "The developer made a new solution with Python, a helpful language."

b. Make sentences clear: Adding a noun phrase can make a computer science sentence easier to understand.

Example: "The group worked together on the project about machine learning."

c. Create variety: Noun phrases make talking about computer science more interesting and less boring.

Example: "The data person found important information from the big dataset with many data points."

16

Transitions

In academic writing, we use words and phrases to connect ideas across sentences and paragraphs, creating coherence and a smooth flow. These linking devices are known as transition words or phrases.

Transition words and phrases help connect ideas between sentences or even paragraphs. They signal relationships between ideas, such as addition, contrast, cause and effect, time, and sequence. Transition words and phrases guide the reader through the text, making it easier to understand and follow.

Categories of transition words and phrases in computer science contexts:

a. Addition: These words show that more information is being added. Examples: also, moreover, furthermore, in addition, besides.

b. Contrast: These words show a difference or opposition between ideas. Examples: however, on the other hand, in contrast, nevertheless, yet, but.

c. Cause and effect: These words show a cause or result of something. Examples: therefore, as a result, because, since, consequently, thus.

d. Time: These words indicate when something occurs or the order of events. Examples: meanwhile, after, before, during, subsequently, then.

e. Sequence: These words show the order or arrangement of ideas. Examples: first, second, third, next, last, finally

Example:

Python is a popular programming language for beginners. However, some programmers prefer using JavaScript for web development.

In this example, "however" is a transition word that connects the two sentences and shows a contrast between the ideas.

Across paragraphs example:

Python is known for its simplicity and readability, making it a popular choice for beginners. Moreover, it has a wide range of libraries and frameworks.

On the other hand, JavaScript is primarily used for client-side scripting in web development, enabling developers to create interactive websites.

In this example, "moreover" is used to add information to the ideas presented in the first paragraph, while "on the other hand" serves as a contrasting transition between the two paragraphs.

Using transition words and phrases helps create clear, coherent, and well-structured texts by connecting ideas across sentences and paragraphs in computer science topics in the English language.

17

Conjunctions

Clauses are groups of words that contain a subject and a verb. Conjunctions are used to join clauses together to create more complex sentences and convey relationships between ideas.

Types of conjunctions: There are two main types of conjunctions used to connect clauses:

a. Coordinating conjunctions: These conjunctions connect independent clauses (main clauses) with equal importance. Common coordinating conjunctions are "for," "and," "nor," "but," "or," "yet," and "so." You can remember them using the acronym FANBOYS.

b. 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."

Conjunctions play a crucial role in creating clear, coherent, and well-structured sentences by connecting clauses in the English language.

18

The passive voice makes sentences in English where the action is more important than who does it. For example:

1. Weights were initialized using He initialization.
2. ANOVAs were conducted to confirm the instrument’s validity.

In computer science academic writing, people often use the passive voice to focus on the process, the results, or the research, not on who did the work. Using passive voice helps make the writing more objective.

Example of process: The software was updated to fix the bug.

Example of results: The test accuracy was improved by 10%.

Example focusing on research: The model was trained using a large dataset.

Using passive voice in this way helps to emphasize the important parts of the research and keeps the focus on the ideas, not the people.

However, not all verbs can be used in the passive voice. Verbs that don't have an object, called intransitive verbs, cannot be made passive. This is because there is no object to move to the subject position in the sentence.

For example, intransitive verbs like "sleep," "arrive," and "laugh" cannot be used in the passive voice because they don't have an object to play the role of subject in the sentence.

So, remember that only verbs with an object, called transitive verbs, can be used in the passive voice.

19

The most frequently used tenses and aspects in computer science writing are the present simple, present perfect simple, and past simple tenses. Here is an explanation for each tense's usage:

Present Simple: This tense is used to discuss concepts that are true now and generally remain true. It helps readers understand fundamental ideas and current developments in computer science.

Example: "A neural network consists of layers of interconnected nodes."

Present Perfect Simple: This tense is used to describe events that began in the past but still have relevance today, often to highlight the significance of previous research in an introduction.

Example: "Researchers of autonomous lane-following have come up with a multitude of techniques."

Past Simple: This tense is used to describe actions completed in the past, typically related to the execution of a project or experiment.

Example: "The team developed the tool and tested its effectiveness using a large dataset."

Using tenses appropriately is crucial when writing a computer science research paper.

20

In computer science academic writing, we use the present simple tense to explain basic concepts, describe the current state of a technology, or discuss widely accepted theories.

Here are some examples related to computer science academic writing:

1. Data is stored in databases.
2. An algorithm solves a problem.
3. A computer processes information.

We use the present simple tense in computer science academic writing for these reasons:

1. To explain general facts that are true in general in computer science. This helps readers understand the basic ideas and theories.

Example: Machine learning models learn from data.

1. To describe current states, properties, features or characteristics of a technology or system at the time of writing. This helps readers understand what is happening in the field of computer science.

Example: A neural network consists of layers of interconnected nodes.

1. To discuss widely accepted theories or principles that are commonly accepted in the field of computer science. This helps readers understand the foundation of the subject.

Example: "Information travels through networks using the Internet Protocol."

In summary, the present simple tense is used in computer science academic writing to explain general facts, describe current states or properties, and discuss widely accepted theories.

21

In computer science academic writing, we use the present perfect simple tense to emphasize the ongoing importance of past developments, achievements, or discoveries. This is often done in the Introduction section when discussing studies related to our topic.

Here are some reasons and situations to use the present perfect simple in computer science academic writing:

1. To highlight the ongoing importance of past developments and show that a past action, development, or discovery is still relevant or significant in the present.

Example: "Researchers have made significant advancements in artificial intelligence."

1. To describe actions that occurred in the past but have a direct impact on the present situation.

Example: "Computer scientists have improved data storage techniques, enabling us to store more information."

1. To describe past events when the exact time is not important, but the event's impact on the present is relevant.

Example: "Scientists have discovered new methods for optimizing algorithms."

In summary, the present perfect simple tense is used in computer science academic writing to emphasize the ongoing importance of past developments, indicate past actions with present results, and describe actions that occurred at an unspecified time in the past. This tense helps convey the connection between past events and their present significance.

22

In the Methods section of computer science academic writing, the past simple tense is used to clearly describe the steps and procedures followed during experiments, studies, or the development of software or algorithms.

Here are some reasons and situations when the past simple tense is employed in the Methods section of computer science academic writing:

1. To outline past actions or procedures: This helps readers grasp the research process.

Example: "We collected data from various sources and preprocessed it to eliminate inconsistencies."

1. To report the utilization of tools or software: This specifically mentions the tools, software, or programming languages employed.

Example: "The researchers utilized Python to develop the machine learning model."

1. To detail the experimental setup: This includes a description of the research participants.

Example: "Participants were divided into two groups, with each group assigned a distinct task."

1. To explain the data analysis process: This encompasses the criteria used to evaluate outcomes.

Example: "Data was analyzed using statistical methods to ascertain the significance of the results."

In summary, the past simple tense is employed in the Methods section of computer science academic writing to describe past actions, procedures, tools, experimental setups, and data analysis.

23

Hedging in academic writing means using words to make your ideas less strong or direct. Some words used for hedging include might, could, may, possibly, seems, and generally.

For example: "Traffic accidents may be reduced by the mass adoption of self-driving vehicles." or "This method could improve performance."

In computer science academic writing, hedging is used because researchers want to be accurate and careful when sharing their findings. This field is complex, and sometimes there are different ways to solve a problem or new information that can change our understanding of things. By using hedging, writers show they are open to other ideas and avoid making statements that are too strong or certain. This helps create a respectful and thoughtful conversation among researchers.

24

Boosters are words that make sentences stronger and more certain. Some common boosters are definitely, clearly, undoubtedly, and always.

For example: "The algorithm clearly improves speed." or "The software is undoubtedly user-friendly."

In computer science writing, boosters are used when the writer feels confident about their ideas or wants to make a point stronger. They help show that the writer believes their results are accurate and can be trusted. However, boosters should be used cautiously and only when the evidence strongly supports the idea. Using too many boosters can make the writing seem overly confident, which may not leave a good impression on the reader.

25

Approximation in English means using words to indicate a number or size without being exact. Words like about, around, nearly, and almost are used for approximation.

In computer science academic writing, approximation is used when we want to convey a general idea without providing the exact number. This is helpful when the precise number is not very important or when simplifying information for better understanding.

For example: "The software takes around 10 minutes to install." or "The system can handle nearly 200 users at once."

Approximation is employed in computer science writing when being highly exact is not necessary. It makes the writing clearer and more straightforward, especially for readers who don't need all the details.