



Research Journalist &
Online Publisher



Guide



Fixing Arabic Text Direction

 **YouTube**



Prepared by: Said Samir
www.ssamir.com

Said Samir

Mastering RTL Content Online

A Technical Guide to Publishing Right-to-Left Languages Like Arabic, Hebrew, and Farsi on Online Platforms

Introduction: The Challenge of RTL Text Online

Publishing content in Right-to-Left (RTL) languages on the internet can be challenging. While many modern platforms have improved their support, users often encounter issues where RTL text (such as Arabic, Hebrew, or Farsi) displays incorrectly, appearing Left-to-Right (LTR) or with jumbled characters, especially when browser or interface settings default to Latin languages. This guide provides technical explanations and best practices to ensure your RTL content is always presented correctly.

Understanding RTL and the Unicode Bidirectional Algorithm (UBA)

Every character in Unicode has a specific directionality. Latin characters have a **Strong LTR** (Left-to-Right) property, while Arabic, Hebrew, and Farsi characters have a **Strong RTL** (Right-to-Left) property. The Unicode Bidirectional Algorithm (UBA) is a complex set of rules that determines the visual ordering of mixed-direction text within a single line or paragraph. Issues commonly occur when mixing scripts or when the base direction of the text or its container isn't clearly defined, leading to misinterpretations by the UBA.

Universal Best Practices for RTL Content

These practices are fundamental and apply across almost all online platforms where you input text.

1. Utilize Unicode Control Characters (UCCs)

These are invisible characters that explicitly guide the UBA, preventing misinterpretations. They are your primary tool for robust RTL rendering.

- **Right-to-Left Mark (RLM):** This character strongly indicates that the following text should be rendered RTL.
 - Character: `` (U+200F)
 - HTML entity: `‏`
 - Direct copy (invisible):

Best Use: Always add RLM at the beginning of any RTL text block, especially descriptions, titles, or comments where you suspect an LTR default context.

هذا وصف باللغة العربية.

This description is in English.

- **Left-to-Right Mark (LRM):** This character strongly indicates that the following text should be rendered LTR.
 - Character: `` (U+200E)
 - HTML entity: `&lm;`

Best Use: Use LRM to clarify LTR segments within an RTL block, particularly around numbers or symbols, to bind them correctly to the LTR content they belong to.

- **Zero-Width Non-Joiner (ZWNJ):** Crucial for languages like Farsi and some Arabic scripts. This character prevents adjacent characters from joining when they normally would, which is essential for correct word formation.
 - Character: `` (U+200C)
 - HTML entity: `‌`

Best Use: Insert ZWNJ where characters should not connect, e.g., in Farsi between a prefix/suffix and the main word to prevent incorrect ligatures.

2. Apply HTML `dir` Attribute

While UCCs handle inline text, the `dir` attribute explicitly sets the base direction for larger blocks of text or the entire page.

- `dir="rtl"`: Apply this to paragraphs (`

`), divisions (`

`), or spans (` `) containing RTL text, especially if they might be nested within an LTR container.

- **`<bdi>` (Bidirectional Isolation):** This HTML element is ideal for mixed-direction content. It tells the UBA to treat its content as isolated from the surrounding text's direction. Use it for user-generated content or where direction might be unpredictable.

```
<p>اسم المستخدم<bdi>John Doe</bdi></p>
```

3. Handling Mixed-Script Text

While it's best to avoid mixing languages in a single text block if possible, sometimes it's unavoidable (e.g., brand names, technical terms).

- **Embed LTR within RTL:** If an English word or phrase is within an Arabic/Hebrew/Farsi sentence, place the LTR part in the middle of the RTL phrase.

شرح لغة Python للمبتدئين
مقارنة بين iOS و Android للمطورين

- **Software/Website Names:** Treat proper nouns as LTR text embedded within the RTL flow.

شرح برنامج Adobe Photoshop بالعربية
طريقة استخدام Google Analytics للمواقع

4. Working with Numbers and Punctuation

- **Arabic-Indic Numerals:** Use localized numerals (٠١٢٣٤٥٦٧٨٩) if your content's audience prefers them. Most platforms will display them correctly if the font supports them.

النتيجة: ٩٧٪ من المشاركين نجحوا

- **Latin Numerals in RTL Text:** If you must use Latin numerals (0123456789) within RTL text, embed them carefully.

هذا هو الجزء رقم ٥ من السلسلة
المدة: ١٥ دقيقة تقريباً

- **Punctuation:** Pay close attention to punctuation marks. They are "neutral" characters in UBA. Add RLM/LRM after punctuation marks to bind them to the preceding text if misplacement occurs.

Platform-Specific Best Practices

YouTube

- **Video Titles & Descriptions:** Always start your Arabic, Hebrew, or Farsi titles and descriptions with an ****RLM**** (```). This forces YouTube's rendering engine to adopt an RTL base direction for that text block, even if the user's interface is LTR.
- **Comments:** Use the RLM trick for comments as well, especially if your comment contains mixed content or is a reply to an LTR comment.
- **Channel Description:** Apply the same RLM rule to your channel's "About" section for consistent display.

Facebook (and other Social Media like X, Instagram)

Similar to YouTube, social media platforms often default to LTR. The RLM character is your best friend here too.

- **Posts & Captions:** Begin your RTL posts and photo/video captions with an ****RLM**** (```). This guides the platform's rendering.
- **Comments:** Use RLM at the start of your comments for correct directionality.
- **Profile/Page Descriptions:** Ensure your "About" sections or bio content also start with RLM if they are primarily in an RTL language.
- **Hashtags:** If using mixed-language hashtags (e.g., `#تقنيةTech`), ensure the RTL part is correct. RLM can sometimes help if the hash symbol misplaces the text.

Common Pitfalls to Avoid

These are common practices that can confuse the Unicode Bidirectional Algorithm:

- **Don't start or end with:**

- **Numbers:** E.g., `الموضوع الأول 1.` (Wrong)
- **Symbols:** E.g., `>> النقطة الأولى` (Wrong)
- **Extra spaces:** Leading or trailing spaces can sometimes interfere with UBA's auto-detection.
- **English text:** E.g., `Title: العنوان العربي` (Wrong)

Instead, use: Start directly with the RTL text and embed LTR content if necessary.

النقطة الأولى في الموضوع
العنوان: أهمية التكنولوجيا

Testing Your Content

Always verify your content before publishing to ensure it displays correctly across different environments.

1. **Copy and paste into a text editor that supports RTL:** Use tools like Notepad++ (with appropriate plugins), VS Code, or even a simple RTL-aware word processor to preview your text.
2. **Verify RTL display:** Confirm that Arabic/Hebrew/Farsi text displays right-to-left, and mixed content is correctly ordered.
3. **Test in different browsers:** Check how your content appears in Chrome, Firefox, Safari, Edge, etc.
4. **Check with different interface languages:** View your content on platforms when the interface language is set to both RTL (e.g., Arabic) and LTR (e.g., English) to see how the UBA reacts.

Helpful Tools and Resources

- **Online RTL Text Fixers:** Websites like [FixTxt.co](https://fixtxt.co) can help automatically add RLM or fix common RTL text alignment issues.
- **Text Editors with Full Unicode Support:** Use modern code editors (e.g., VS Code) that have excellent Unicode and bidirectional text rendering capabilities.
- **Browser Extensions:** Extensions like "RTL Toggle" (for browsers like Chrome/Firefox) can temporarily switch the text direction of a webpage, useful for debugging.

- **Unicode Consortium:** Refer to the official Unicode Standard Annex #9 for the full Unicode Bidirectional Algorithm specification (for advanced understanding).

Key Takeaway

Successfully displaying Right-to-Left text correctly online, whether it's Arabic, Hebrew, or Farsi, depends on understanding the fundamental principles of the Unicode Bidirectional Algorithm and consistently applying appropriate control characters and semantic HTML. It's not about luck or trial-and-error, but a structured approach to text management. This proactive method ensures your content reaches your audience as intended across diverse digital platforms.

This guide is prepared based on W3C and Unicode Consortium standards for bidirectional text.

Prepared by Said Samir www.ssamir.com ©2025 All rights reserved.