AI-based languages for non-existent | gibberish words in WProofreader

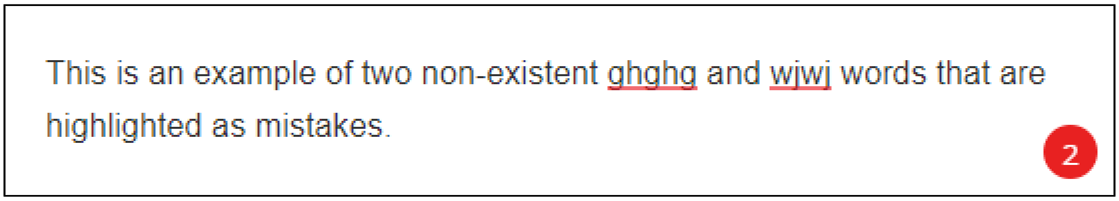
AI-based engine was trained to work with **real** text and skip non-existent wording, that is usually random combinations of letters typing like words in haste for testing purposes. By default, the AI-based engine ignores such nonsense wording and doesn't highlight them as mistakes or provide any suggestions for them even if they are in the middle of the real text.

> ⚠ The default language for our products is 'auto', which applies AI-based proofreading for text once the English language is identified. This means that 'auto' language will not provide suggestions for non-existent words as it will use the AI-based engine under the hood for English texts.

> ⓘ Unlike AI-based engines, classic spell checking is very “simple” (e.g. American English), it relies on the wordlist. And if the word is not in the dictionary, it will be considered a misspelt word. In that case, such a word would be underlined since the spell-check mechanism is built on top of the wordlists. Thus, when you switch to a dictionary-based language, you can see the suggestions for non-existent words.

In 2023, we have taken significant steps to address the case of nonsense words in the WebSpellChecker release 5.28.0.0. With the introduction of AI English, our advanced language processing technology, we now offer an improved solution for identifying gibberish or unrecognized words.

In this update, WebSpellChecker incorporates a powerful mechanism that goes beyond traditional spell-checking capabilities. Whenever your text contains nonsensical words, our system automatically scans them and compares them against our extensive dictionary. If any of these words are not found in the dictionary, they are promptly highlighted with a red underline, without providing suggestions not to confuse your end-users.



This is an example of two non-existent ghghg and wjwj words that are highlighted as mistakes.

The purpose of this feature enhancement is to provide you with a clearer indication of words that may require further attention or clarification. By visually highlighting unrecognized or potentially incorrect words, we aim to empower you in ensuring the accuracy and integrity of your written content.

Note, if you want to highlight the nonsense words you have to take some steps for updates on your side. For on-premise version work with gibberish words can be configured for AI-based, auto or classic languages with using additional enforceAI parameter. For Cloud version, only two last options works and the configuration will be the same for both versions.

I. On-premise version 5.28.0.0+ for AI-based language:

> ⚠ Configuration below is the example for AI-based English (en_AI) in combination with American English (en_US). You can configure the same for AI-based German (de_AI) or AI-based Spanish (es_AI) and appropriate classic languages' pares.

1) Stop AppServer

2) Navigate to **AppServerX** file. By default it is located at WebSpellChecker_Installation_path\WebSpellChecker\AppServer

3) Modify the SpellCheck Enabled and SpellCheckPriority values and add DisableSuggestions tag in the <Language Id="en_AI" Enabled="true"> tag :

```
<SpellCheck Enabled="true" DisableSuggestions="true">
    <Locale>en_US</Locale>
    <Provider>hunspell</Provider>
    </SpellCheck>
<SpellCheckPriority>true</SpellCheckPriority>
```

4) Start AppServer

II. Cloud and On-premise versions for AI-based language working under the hood:

a) When AI-based language works under the hood of the 'auto' language. Thus, the following configuration is needed in your WEBSPELLCHECKER_CONFIG:

```
lang: 'auto',
enforceAI: true,
autoLangPriorities: {
en: 'en_US',
},
```

b) When AI-based language works under the hood of the classic engine. Thus, the following configuration is needed in your WEBSPELLCHECKER_CONFIG:

```
lang: 'en_US',
enforceAI: true,
```