### **1. Build a Dictionary-Based Word Bank**

* Use a dictionary (e.g., WordNet, Wiktionary) to collect all known words.
* Categorize words into **grammatical groups** (nouns, verbs, adjectives, etc.).
* Store this data in a structured format (e.g., JSON, SQLite, or a Trie).

### **2. Implement Grammar-Based Prediction**

* Analyze sentence structure to determine what word category is expected next.
* Example: If the user types **"The cat"**, the next word is likely a **verb**.
* Use a simple **Part-of-Speech (POS) tagging model** (e.g., spaCy or NLTK) to analyze input context.

### **3. Rank Predictions by Priority**

* **Word Popularity:** Use frequency counts from corpora like Google N-grams or Common Crawl.
* **Context Matching:** Give higher priority to words that fit semantically.
* **User Preferences:** Adapt based on past user input.

### **4. Implement Word Autocomplete**

* Use a **Trie data structure** to quickly suggest word completions based on prefixes.
* Combine with your grammatical analysis to refine results.

### **5. (Optional) Improve Accuracy with ML**

* Train a small **language model** (LSTM or Transformer) on labeled grammatical structures.
* Use it to **validate** or **re-rank** predictions.