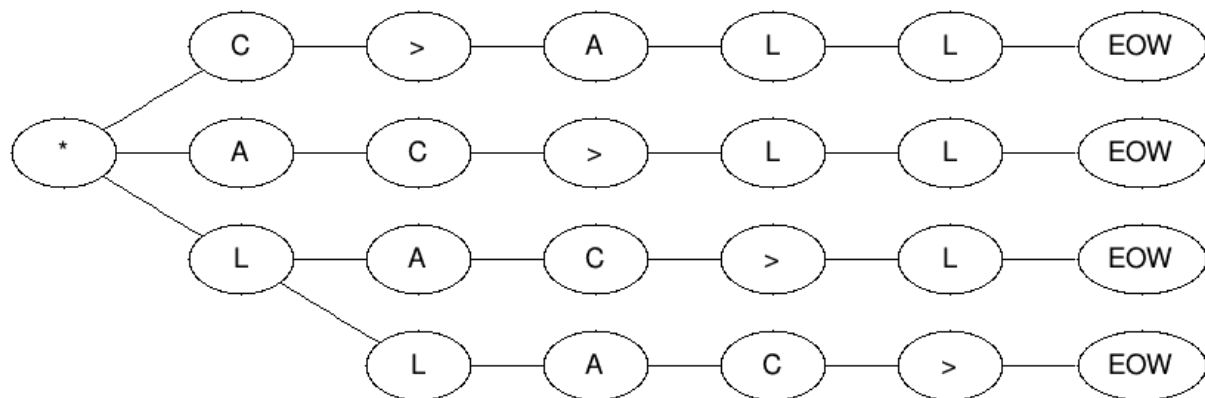
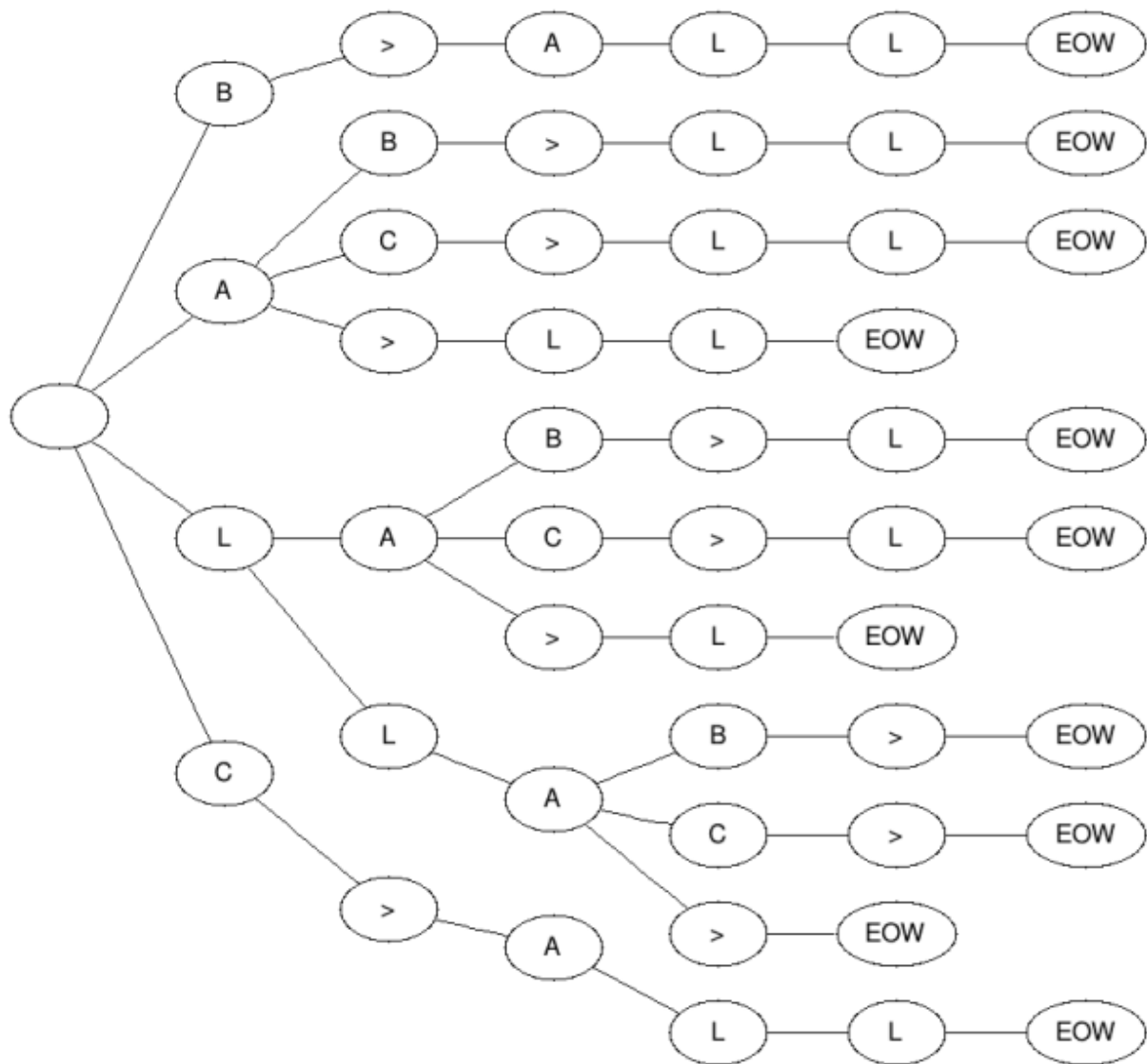


GADDAG is a data structure proposed by Steven Gordon that is optimized for searching for words in games like Scrabble and Words With Friends. It's similar to a Trie prefix tree, however it's organized so that all "hooks" (letter sequences on the board) are available from the root node. This is done by storing every possible prefix and suffix in the word, with the prefix being reversed off of the root node and two parts being separated by a node typically represented by the ">" character.

For instance the word "Call" stored in a GADDAG looks like this:



The magic of this structure is if you want to check if a letter string exists all you need to do is reverse the string and check the branch starting from the first letter. For instance if you had a structure that contained "Call", "Ball", and "All":



You can find all words with “L” in them by taking the “L” from the root then crawling down all the branches. Beyond a single letter you can look for words that contain “ALL” and drill down the nodes in reverse by transversing L->L->A.

One of the downsides to this structure is it takes up a lot of memory, however the paper above gives instructions on how to perform a partial compression of the nodes by reintegrating them after the break, as seen here:

