Query Expansion Brain storm

Query Expansion, as the name suggests is expanding the user’s original query using any of the following techniques:

1. **Synonym Expansion**- Add synonyms (regular or semantic) of some or all of the user’s query terms
    a. What constitutes a breach of contract in florida? - Can be expanded using words like agreement, lease, violation, elements of, etc.
    b. These synonyms can be English based (contract vs agreements) or Legal based (Fingers vs Digits)
    c. Words could be more general hypernyms(umbrella terms) or specific hyponyms
        i. E.g Add automobile if the word car is mentioned in the query and car if automobile was used in the query
    d. Can improve both Precision and Recallbased on the quality of expansion
        i. Replacing/Expanding with correct Legal Terms can improve precision
            1. E.g Replace Fingers with Digits or Add Digits
        ii. Recall can be increased by using hypernyms or co-hyponyms, sometimes at the cost of precision
            1. E.g. When a particular brand of car is mentioned in the query, and we expand to other brands could increase recall as they are all cars, but reduce the precision
2. **Stemming/Lemmatization**- Is also a type of query expansion, where specific words are replaced with their general/root counterparts
    a. Harmless cases - Duty matches duties
    b. Possibly harmful - Operate | Operating | Operates | Operation | Operative | Operatives | Operational - All are stemmed the same way.
        i. A query about doctors doing an operation, we could potentially match on all the cases about any one operating any equipment or any operating system like unix or windows

c. Mostly improves Recall
d. Already exists in our system

Query Expansion – Expand some of all of the user query terms using their synonyms or relevant words.

Interface considerations exist - Display the expansion to the user

Query expansion terms are typically abbreviations or synonyms. (?)

The best — or at least most principled — approach is to integrate query expansion into a machine learned ranking model using features (in the machine learning sense) that indicate whether a document matched the original query terms or terms introduced through query expansion. These features should also indicate whether the expansion was through an abbreviation or a synonym, the similarity of the synonym, etc.

Integrating query expansion into a machine-learned ranking model is a bit tricky. We can’t take full advantage of pre-existing training data from a system that hasn’t performed query expansion. Instead, we start with a heuristic model to collect training data (e.g., one of the previously discussed approaches) and then use that data to learn weights for query expansion features.