**1. Approach for extracting word pairs and ranking**

**It was done by finding candidate word pairs of certain POS, filter them with possible intensifiers, and ranking them with frequency.** Possible POS pairs are ‘adv’+’adj’(e.g. deathly sick) and ‘adj’+’n’(e.g. stark contrast). In the second case, since the adjective has to intensify the meaning of the noun, nouns are limited to ‘abstract noun’. However, in this approach, there is no way to distinguish abstract nouns from concrete nouns, only the nouns end with ‘-ness’ are extracted. Then, they were filtered with some criteria. If w2 in (w1,w2) are appearing more than once, than the probability that w1 may not intensify w2 is bigger than the opposite, so discarded. If w1 appears more than 50 times in the whole corpora, than it is not likely to be intensifiers (e.g. ‘soon’), or more general intensifiers(e.g. ‘greatly’), so discarded. Also, since we are finding intensifiers for limited group of lexicons, w1 used with more than 3 lexicons are also discarded. Making conditional frequency distribution dictionary with these pairs, the uniqueness was evaluated. For each possible word pairs (w1, w2), uniqueness was evaluated as freq(w1)/freq(w1, w2) was measured and used in ranking.

**2. Overall quality of the output**

Among hundred outputs, 79 of them are treated as proper outputs. Among them, 41 contained intensifiers with ‘similar meaning’ with the next lexicon, so intensifying the meaning (e.g. cat-like sinuousness) and 38 of them were ‘general intensifiers’ to limited lexicons (e.g. marvelously high-pitched). 17 of them were not an intensifier (e.g. ecologically fragile). For one of them, the POS finding was bad that nouns with ‘-ness’ return some concrete nouns (e.g. epileptic governess). For two of them, I couldn’t find the words’ meaning in the dictionary (e.g. garpian randomness).

Ranking was good, too. When I classify those 79 ‘proper’ outputs, there were two classes, A for ‘similar meaning’ pairs and B for ‘general intensifier’ pairs, as explained above. Among 28 proper ‘uniqueness 1’ outputs, 19 were class A and 9 were class B. Among 22 proper ‘uniqueness 0.5’ outputs, 10 were class A and 12 were class B. Among 16 proper ‘uniqueness 0.33’ outputs, 8 were class A and 8 were class B. Among 13 proper ‘uniqueness 0.25’ outputs, 4 were class A and 9 were class B. This result is consistent with the idea that ‘similar meaning’ pairs have to be more limited to decorate other lexicons than the ‘general intensifiers’.

**3. limitations and further improvement.**

This approach has innate **limitation in recognizing the intensifiers**. In this approach, I assumed that if the second lexical item is decorated by more than one lexical item (not including very, since it is a stopword), then the first lexical item is not necessarily an intensifier. For example, if (minimally, necessary) and (really necessary) appears in the list, minimally is not likely to used as an intensifier, so both of them are discarded. However, since some lexicons can be intensified by several intensifiers, this approach deletes many of the possible outputs.

Also, I need **more tagged corpus** for better output quality. Right now, there are many output pairs that appears only once in the input corpora. It might work if I use a tagger to tag some untagged texts. Thus the approach for filtering non-intensifier words and ranking does not really work.

**Ranking** is not perfect, too. For example, ‘sound’ in ‘sound asleep’ is not always used to decorate other lexical items. It is more often used as a verb. However, ‘asleep’ is almost the only word which ‘sound’ decorates. This can be solved by using more tagged corpus as an input and replace the fdist used for calculating frequency of w1 to cfdist. (In this manner, we can only count the words used to decorate the other lexicon.)

Also, since I didn’t **extract all the abstract nouns** for ‘adj’+’n’ pairs, there might be more possible word pairs. If there is a good way to extract the abstract nouns, we can get more outputs. Using some semantics of abstract nouns will help. For example, ‘-cy’, ‘-ness’, ‘-ity’ are nouns that are made by adjectives, so abstract nouns. Also, concrete nouns are often used as subject, so discarding the lexicons used as subjects may help finding abstract nouns.

There is also a new approach I can try. This is **using some general intensifiers for the input**. For example, search for ‘very’+’adj’ or ‘highly’+’adj’ and find ‘adv’+’adj’ pairs that are used in the same context with those word pairs. Ranking with same approach would work.