Q0

a) I'm not sure whether I should print the max depth or the relative depth, so both are here:

```
Deepest Synset: Synset('rock_hind.n.01')
Maximum Depth: 19
Hypernym Paths and Depths:
Path 1:
  Synset('entity.n.01') (Depth: 0)
  Synset('physical_entity.n.01') (Depth: 1)
  Synset('object.n.01') (Depth: 2)
  Synset('whole.n.02') (Depth: 3)
  Synset('living_thing.n.01') (Depth: 4)
  Synset('organism.n.01') (Depth: 5)
  Synset('animal.n.01') (Depth: 6)
  Synset('chordate.n.01') (Depth: 7)
  Synset('vertebrate.n.01') (Depth: 8)
  Synset('aquatic_vertebrate.n.01') (Depth: 9)
  Synset('fish.n.01') (Depth: 10)
  Synset('food_fish.n.01') (Depth: 11)
  Synset('sea_bass.n.02') (Depth: 12)
  Synset('grouper.n.02') (Depth: 13)
  Synset('hind.n.01') (Depth: 14)
  Synset('rock_hind.n.01') (Depth: 15)
Path 2:
  Synset('entity.n.01') (Depth: 0)
  Synset('physical_entity.n.01') (Depth: 1)
  Synset('object.n.01') (Depth: 2)
  Synset('whole.n.02') (Depth: 3)
  Synset('living_thing.n.01') (Depth: 4)
  Synset('organism.n.01') (Depth: 5)
  Synset('animal.n.01') (Depth: 6)
  Synset('chordate.n.01') (Depth: 7)
  Synset('vertebrate.n.01') (Depth: 8)
  Synset('aquatic_vertebrate.n.01') (Depth: 9)
  Synset('fish.n.01') (Depth: 10)
  Synset('bony_fish.n.01') (Depth: 11)
  Synset('teleost_fish.n.01') (Depth: 12)
  Synset('spiny-finned_fish.n.01') (Depth: 13)
  Synset('percoid_fish.n.01') (Depth: 14)
  Synset('serranid_fish.n.01') (Depth: 15)
  Synset('sea_bass.n.02') (Depth: 16)
  Synset('grouper.n.02') (Depth: 17)
  Synset('hind.n.01') (Depth: 18)
  Synset('rock_hind.n.01') (Depth: 19)
```

```
Path 1:
  Synset('entity.n.01') (Depth: 0)
  Synset('physical_entity.n.01') (Depth: 1)
  Synset('object.n.01') (Depth: 2)
  Synset('whole.n.02') (Depth: 3)
  Synset('living_thing.n.01') (Depth: 4)
  Synset('organism.n.01') (Depth: 5)
  Synset('animal.n.01') (Depth: 6)
  Synset('chordate.n.01') (Depth: 7)
  Synset('vertebrate.n.01') (Depth: 8)
  Synset('aquatic_vertebrate.n.01') (Depth: 9)
  Synset('fish.n.01') (Depth: 10)
  Synset('food_fish.n.01') (Depth: 11)
  Synset('sea_bass.n.02') (Depth: 16)
  Synset('grouper.n.02') (Depth: 17)
  Synset('hind.n.01') (Depth: 18)
  Synset('rock_hind.n.01') (Depth: 19)
Path 2:
  Synset('entity.n.01') (Depth: 0)
  Synset('physical_entity.n.01') (Depth: 1)
  Synset('object.n.01') (Depth: 2)
  Synset('whole.n.02') (Depth: 3)
  Synset('living_thing.n.01') (Depth: 4)
  Synset('organism.n.01') (Depth: 5)
  Synset('animal.n.01') (Depth: 6)
  Synset('chordate.n.01') (Depth: 7)
  Synset('vertebrate.n.01') (Depth: 8)
  Synset('aquatic_vertebrate.n.01') (Depth: 9)
  Synset('fish.n.01') (Depth: 10)
  Synset('bony_fish.n.01') (Depth: 11)
  Synset('teleost_fish.n.01') (Depth: 12)
  Synset('spiny-finned_fish.n.01') (Depth: 13)
  Synset('percoid_fish.n.01') (Depth: 14)
  Synset('serranid_fish.n.01') (Depth: 15)
  Synset('sea_bass.n.02') (Depth: 16)
  Synset('grouper.n.02') (Depth: 17)
  Synset('hind.n.01') (Depth: 18)
  Synset('rock_hind.n.01') (Depth: 19)
```

Q1

d)

The extension is helpful since it incorporates the definition of not only the word itself, but also words similar to it. Thus, the extended Lesk algorithm increases the potential for overlap with the context. Since the algorithm is based on maximizing overlap, a larger signature increases the chances that the correct sense will have a higher overlap score than incorrect ones.

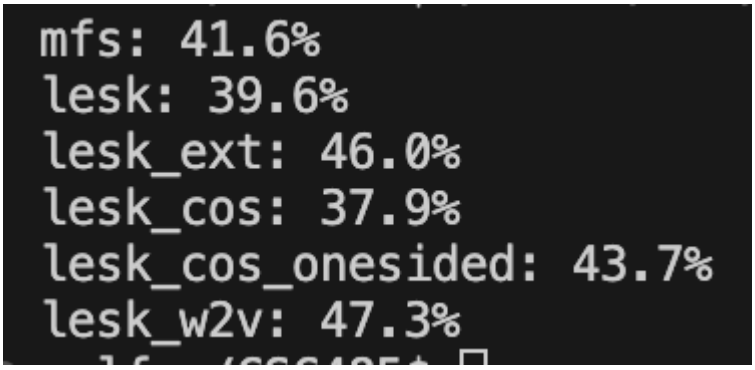
g)

Lesk_cos_oneside performs better. I believe this is because there's less distraction for words that have multiple meanings. In other words, the signature may be too generic and thus provide noise to the algorithm. For example, for a sentence like "a dog drinks water near a bank", clearly the word "bank" refers to riverside. But if the signature consists of the financial institution meaning of "bank", the algorithm may get confused.

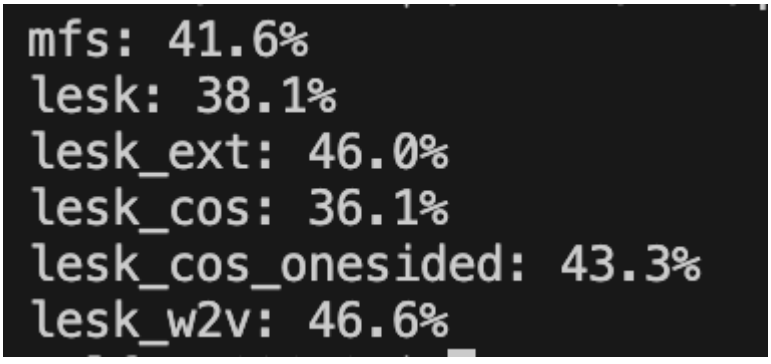
h)

The numerator is the dot product of the sets, which counts the number of dimensions where both vectors have a 1. This is the same as the intersection of the sets. The denominator is the product of the norms. The CosSim would still be greatest if the sets are the same, and would be 0 if none of the elements in the set is the same.

j)
normally:



With lower cases:



The accuracy drop slightly for all methods except mfs and lesk_ext. The reason may be that mfs only takes the max frequency and does not look at the meaning of the word, and the lesk_ext has broad enough signatures to be able to get the correct meaning of the word. The other ones are affected by losing the distinct meaning of capitalized words, for example the “Apple” company and the fruit “apple”.

Q2
a)
Yes, context is necessary. There are several cases where the algorithms implemented in Q1 may be unable to distinguish the meaning.

1. Same word order, but different meaning: words may have different meanings in the 2 sentences even if they are in the same position. Similar sentences include: “he was hungry so he ate the duck.” and “I saw a ball coming so I duck.”, where in the first sentence “duck” is a noun referring to animal, in the second sentence it describes the action. The algorithms can not differentiate because they only look at the order of the word, instead of the context.

2. Multiple meanings of word itself: examples include “I saw the bank”, where "bank” may refer to a river shore or a financial institution. With only this sentence, algorithms could only guess the meaning of the word.
3. Order matters: when there is “no dependencies, semantic roles, parts of speech”, a sentence’s meaning may be totally different. E.g. “The cat ate a fish”, if only looking at each word, there’s no ambiguity, but without the ordering information, the sentence may become “The fish ate a cat”.
4. Like the provided example, “play” may have different meaning based on context, but w2v would assign the same vector.

All of the above are what may happen to the algorithm. But in general, order-invariant models are more sensitive to words that have different meanings depending on their position and the context provided by surrounding words. The reason is that word order-invariant methods do not consider sequential information.

c)

Because of the need for padding. If the longest corpus is at the beginning, the corpus later would only need to pad to the maximum length. If there are shorter corpus at first and longer ones at last, for each longest corpus newly found, all previous corpus would need to be re-pad to align the length.

e)

For lesk variants in Q1 there may be another problem when disambiguating arbitrary sentences. To figure out the correct meaning, the lesk algorithm depends on the synsets of the word. If the word is not in the database, then it’s impossible to decide it’s meaning. In the real world, especially online, many words have abbreviations, or may be misspelled intentionally as in memes. Or the word may have other meanings (metaphoric or sarcastic) than the recorded meaning.

CSC 485H/2501H, Fall 2023: Assignment 2

Family name: Zhai

Given name: Xueqing

Student #: 1006962413

Date: Nov 3, 2023

I declare that this assignment, both my paper and electronic submissions, is my own work, and is in accordance with the University of Toronto Code of Behaviour on Academic Matters and the Code of Student Conduct.

Signature: Xueqing Zhai