Machine Learning for NLP – Assignment 2

Part 1 – Model Training

Preprocessing Steps (in order):

- ***Lowercase text***: We decided not to have different embeddings for lower-case and upper-case text and converted our text to lowercase. This helps us reduce the vocabulary. Generally, word is upper-cased when it's the first word starting the sentence, or it represents someone's name. In both cases, we don't need to distinguish between lowercase and uppercase words. Therefore, we decided to lowercase the text.
- ***Splitting the sentences***: As the context between a new sentence and previous sentence usually differs, splitting the sentence makes sure that wrong context words are not included in the training data.
- ***Punctuation based tokenization***: Using nltk punctuation tokenizer, we further split the text. Including punctuations can replace a possibly correct context word, and thus we decided to remove punctuations.
- ***Not removing stopwords***: We decided not to remove stopwords as it provides significant semantic meaning to the text.

Class Structure

```
class CBOW(nn.Module):

    def __init__(self, vocab_size, embedding_dim, hidden_dim=256):
      super(CBOW, self).__init__()
      # Embedding layer - Lookup table
      self.embeddings = nn.Embedding(vocab_size, embedding_dim)
      # Layer 1 - Since we'll be summing up the context vectors,
      # the input to this layer will be embedding_dim. Output is a 256 dim. vector
      self.linear1 = nn.Linear(embedding_dim, hidden_dim)
      # Adding non-linearity through ReLU
      self.relu = nn.ReLU()
      # Final layer to get to vocab size dim
      self.linear2 = nn.Linear(hidden_dim, vocab_size)
      # Log softmax to get probabilities
      self.log_softmax = nn.LogSoftmax(dim=-1)

    def forward(self, inputs):
      # inputs are context vectors. Get embeddings for them
      x = self.embeddings(inputs)
      # sum all context vectors
      x = torch.sum(x,axis=1).view(inputs.shape[0],-1)
      #x = sum(x).view(1,-1)
      # Add first layer
      x = self.linear1(x)
      # Add relu
      x = self.relu(x)
      # Add final layer
      x = self.linear2(x)
      # Get log softmax
      x = self.log_softmax(x)
      return x
```

- ***Embedding Layer***: Embedding layer to transform the integer tokens to vectors (of length 50).
- ***Summing up the context vectors***: As indicated in CBOW formula, we sum up the context vectors using ***torch.sum***, and reshape the vector considering batch size.
- ***Hidden Layer***: Hidden layer for the neural network. We use 256 nodes in the hidden layer.
- ***Non-linear Activation***: We use ReLU activation to add non-linearity.
- ***Predicting target word***: We project the vectors from hidden layer to vocabulary size followed by softmax activation to generate the probability distribution over the vocabulary. The max probability will be used as predicted target word.

Part 2 – Model Testing

Methodology – For testing the trained word embeddings, we retrieved closest neighbours using two methods. First, we simply retrieve the top 5 neighbours through the logic mentioned in the assignment. In addition, we also tried to relative frequency thresholding of the candidate neighbor words to limit the top 5 neighbours with low frequencies. In the second method, we consider the candidate word only if it's frequency in the vocabulary is at-least 25% of the input word. This is based on the hypothesis that as low frequency words are poorly represented in the training data with very few instances, they might have not been trained properly.

Trip Advisor Results

Input Word	Model	Closest Words	Closest Words (frequency thresholding)
hotel (noun)	CBOW2	daythere, fakes, antibacterial, averge, ptns	nice, room, not, stay, great
hotel (noun)	CBOW5	smirnoff, you, centeredness, tallers, frescos	nice, room, just, good, did
room (noun)	CBOW2	floormat, unhurried, combination, grilling, hostage	clean, no, good, location, time
room (noun)	CBOW5	raffles, overlooking, transferred, creamsickle, quirkiness	did, night, clean, great, rooms
bridge (infrequent noun)	CBOW2	bar10, hustled, congenial, combination, fielded	combination, typically, faced, ran, corridoor
bridge (infrequent noun)	CBOW5	budgeting, persistently, seating, forecourt, tania	seating, steep, place, whilst, gran
booked (verb)	CBOW2	degrading, towards, bordeux, theregood, thorough	business, book, restaurants, prices, bring
booked (verb)	CBOW5	mil, breeding, peaked, becausewe, soilded	use, pretty, mini, business, place
arrived (verb)	CBOW2	alll, graduates, friendley, overseeing, firewood	corner, plaza, use, comfortable, taxi
arrived (verb)	CBOW5	pescatore, garrett, stuffthe, 300k, rennaissance	nearby, loved, located, dont, like
finding (infrequent verb)	CBOW2	earth, perceive, amazedat, task, 24th	switched, e, presented, 120, arrival
finding (infrequent verb)	CBOW5	chocolate, surprised, somone, community, nevis	basket, them, china, tried, wharf
great (adj)	CBOW2	soooooo, maintainence, substituted, corner, dispassionate	stay, helpful, clean, time, nice
great (adj)	CBOW5	tore, detour, harmful, lagoon, scubaed	place, room, bar, rooms, did
helpful (adj)	CBOW2	modeled, noise, batur, witch, undervalued	noise, things, 7, time, drinks
helpful (adj)	CBOW5	discernign, vith, chose, routines, piano	dinner, left, business, felt, park
majestic (infrequent adj)	CBOW2	tourism, hours, encouraged, herman, barcelona ç é	hours, rate, sunday, rates, golf
majestic (infrequent adj)	CBOW5	chic, phony, raffles, nestor, loudly	chic, clubs, mini, past, kitchen

From the above table, we see that:

- The retrieved neighboring words makes sense only for a few instances (room: overlooking, great: harmful) when we don't consider frequency thresholding. However, for frequency thresholding, the results make sense most of the times (hotel: nice, room, stay, great ; majestic: chic).
- Another noticeable observation is that the nearest neighbor retrieval for infrequent retrieval doesn't perform as well as it does for frequent words.
- With relative frequency thresholding, the model performance is decent. However, without frequency thresholding, the model does not perform well. One could argue that words with low frequency could have been removed in preprocessing, but that would come at the risk of mislabeling instances and reducing the training data. Alternatively, longer training period should also improve the model performance.

Sci-fi Results

Input Word	Model	Closest Words	Closest Words (frequency thresholding)
man (noun)	CBOW2	ytccuuough, loiew, ofmush, straps, helmet	without, still, where, another, mind
man (noun)	CBOW5	hejd, headstart, matriarch, looung, triumphantly	around, about, there, great, off, he
eyes (noun)	CBOW2	andblack, auschwitz, hellstorm, tocs, sting	about, even, so, both, what
eyes (noun)	CBOW5	mati, felony, skylarking, rookies, ariel	used, read, five, almost, control
president (infrequent noun)	CBOW2	unremarkably, ncy, rarer, hipster, vation	rogue, spent, tapes, gree, next
president (infrequent noun)	CBOW5	ginwidlah, kish, tarried, annex, arizona	pieces, brass, station, her, dragged
said (verb)	CBOW2	psychopathological, teleportations, recordsi, walkdown, tepeni	this, here, through, s, ?
said (verb)	CBOW5	drfflif, urigiaaf, archaeologists, beluthahatchie, jmfe	don, ?, his, about, an
looked (verb)	CBOW2	vesix, danny, ecstatic, graypec, mowers	almost, d, this, science, work
looked (verb)	CBOW5	sphere, anthropology, otherness, superintend, regenerative	not, part, fiction, wanted, called
eat (infrequent verb)	CBOW2	maghelp, himwas, mechounits, belches, comrades	landed, b, quiet, glass, attacked
eat (infrequent verb)	CBOW5	klars, swallow, justin, comeback, ostentatiously	capable, never, wrote, battle, interesting
good (adj)	CBOW2	skunki, philosopher, transfusion, pfestige, thirtyfifth	feel, except, something, began, yet
good (adj)	CBOW5	sfmagazine, hurrah, eightyfive, emulation, whisper	around, years, both, more, any
old (adj)	CBOW2	tetraploid, grittylooking, kited, puglike, inflation	keep, might, this, s, off
old (adj)	CBOW5	toggle, affrays, tater, drillings, rehearse	number, beyond, not, fact, found
poor (infrequent adj)	CBOW2	eor, carbines, highshouldered, foraging, lungsful	atomic, whisper, bars, color, loaded
poor (infrequent adj)	CBOW5	blundy, freddi, gordinis, chemistical, frog	instrument, neat, shape, political, lonely

From the above table, we see that:

- The results from frequency thresholding does make sense in some cases, but not as many.
- As observed in hotel reviews results, the embeddings with frequency thresholding (eyes: read, president: rogue) are better compared to embeddings without frequency thresholding, and nearest neighbours for infrequent words are worse than nearest neighbours for frequent words.
- A noticeable observation is that embeddings with CBOW5 are better than embeddings with CBOW2 (man: he, looked: called)

Overall, our results on hotel reviews perform better than scifi. Our hypothesis is this happens because of the low number of epochs used while training the scifi model. Although scifi model has more training instances, 3 epochs might not have been sufficient for model training since we did not remove stopwords to retain semantic structure. Increasing the number of epochs might increase the model performance. An alternative strategy for training would be to create a balanced dataset through subsampling.

Comparison of same words across Trip-Advisor and Sci-Fi

Input Word	Dataset	Model	Closest Words (frequency thresholding)
festive	Sci-fi	CBOW2	f, macnessa, waxen, dreadnaughts, somethink
festive	Hotel reviews	CBOW2	itchy, briefing, argumentative, definelty, infallible
festive	Sci-fi	CBOW5	sluiced, microcircuits, collaborating, commended, pauli
festive	Hotel reviews	CBOW5	dismayed, shinkansen, hoards, minding, critiques
incredible	Sci-fi	CBOW2	cone, hue, gallery, alan, hawkins
incredible	Hotel reviews	CBOW2	round, change, pillow, properties, paid
incredible	Sci-fi	CBOW5	plate, absorbed, web, mustache, depend
incredible	Hotel reviews	CBOW5	nicest, affinia, place, ultra, feb

Both the words have different neighbours for sci-fi and hotel review dataset. This is because both the datasets are very domain specific. 'Incredible' or 'Festive' does not imply the same meaning in sci-fi and hotel reviews. As we can see from the results, 'incredible' in hotel reviews can be related to 'place', and 'incredible' in sci-fi dataset can be related to 'mustache'.

Differences between CBOW2 and CBOW5

CBOW2 and CBOW5 are different because the target word is predicted using 4 and 10 context words respectively. This has an impact on the final embeddings of the model. Generally, using more context words implies we can find more relevant words for the target word (only until we don't start including garbage words). This helps in getting more meaningful embeddings for the target word. As we saw in our results, the embeddings from CBOW5 are generally more relevant than embeddings from CBOW2.