PRECOG INDUCTION TASK

HARSHAN N
NATIONAL INSTITUTE OF TECHNOLOGY
THIRUCHIRAPPALII

TABLE OF CONTENTS

PARTS OF THE TASK COMPLETED

I worked on the Language representations task which was comprised of two parts, namely Dense representations and Cross-Lingual Alignment. Along with this, I have a Paper-reading task.

I have completed all the parts of the part-1 and part-2 tasks, starting with the construction of a co-occurrence matrix of the size NxN where I chose the value of N to be equal to 50000. I converted this co-occurrence matrix to sparse matrix as there will be a lot of ‘0’ values in my matrix. Then, I did the task of dimensionality reduction by converting the matrix to an Nxd matrix. I compared this with various values of d and found out that the optimal value of d is ‘7’. I did this using cosine similarity and visualized these embeddings using techniques like SimLex-999, UMAP, and PCA. Finally, I used the pre-trained neural word embedding methods like word2vec and GloVe techniques and compared the co-occurrence counts from both methods. On this comparison found out that our co-occurrence matrix has better co-occurrence counts than these models.

In part-2 task, I took the IIT Bombay’s English-Hindi Bilingual Corpus and pre-processed that file. Then I aligned the embeddings of English and Hindi using the Procrustes analysis method. Finally, I assessed these cross-lingual aligned embeddings using the cosine similarity scores before and after the alignment and visualized it with the help of the PCA method.

Due to time constraints, I was unable to explore alternate cross-lingual alignment techniques like Canonical Correlation Analysis (CCA) and the bonus tasks. I was also unable to work on the multilingual embedding models like MUSE.

METHODOLOGIES

Part-1: Dense Representations

- **Corpus Processing:** I started this task by loading the text corpus from the link provided in the task which was of the size 300k. The sentences from the text corpus were extracted and tokenized the sentences. Following this, a co-occurrence matrix was created and converted to a sparse matrix.
- **Word Embedding Generation:** I constructed this co-occurrence matrix for various window sizes. After that, I applied Singular Value Decomposition (SVD) to reduce dimensions. Further, I computed the results of co-occurrences between the words using cosine similarity and clustering embeddings and visualised this with different methods like SimLex-999, UMAP and PCA.
- **Comparison with Neural Methods:** Finally compared the results with existing neural methods like Word2Vec and GloVe. On comparing we found better results using traditional statistical methods than the neural methods.

Part-1: Dense Representations

- **Preprocessing of Parallel Corpus:** I downloaded the English-Hindi parallel sentences from the IIT Bombay BI-Lingual Corpus. I extracted a dictionary from this corpus and tokenized each sentence as a dictionary after stemming them.
- **Extracting Pre-Trained Word Embeddings:** I extracted the FastText pre-trained embeddings by Wikipedia for both English and Hindi Languages. After extraction, these words were used as word vectors in the bilingual dictionary.
- **Cross-Lingual Alignment:** The transformation matrix was computed using Procrustes analysis and the Hindi embeddings were aligned to the English embedding space.
- **Evaluation of Cross-Lingual Alignment:** The alignment was evaluated using cosine similarity for both before and after the alignment. I was able to see a significant improvement in the cosine values for the words of similar meaning after the alignment

RESULTS

Cosine similarity after SVD Method

d values/word pair	Car & Engine	King & Queen
d=100	0.8735	0.9838
d=200	0.8100	0.9740
d=300	0.7712	0.9667

Spearman Correlation after SimLex-999 Method

d values	Spearmann
d=100	0.0562
d=200	0.0622
d=300	0.0727

RESULTS

Spearman Correlation after UMAP Method

d values	Spearmann
d=100	0.0035
d=200	-0.0010
d=300	0.0005

From the above tables we find that the optimum value for d is 100.

Cosine Similarity for neural Embedding Models

Model	Car & Engine	King & Queen
Word2Vec	0.4080	0.6511
GloVe	0.4842	0.6336

From the above table we find that the traditional model works better as compared to these neural models.

RESULTS

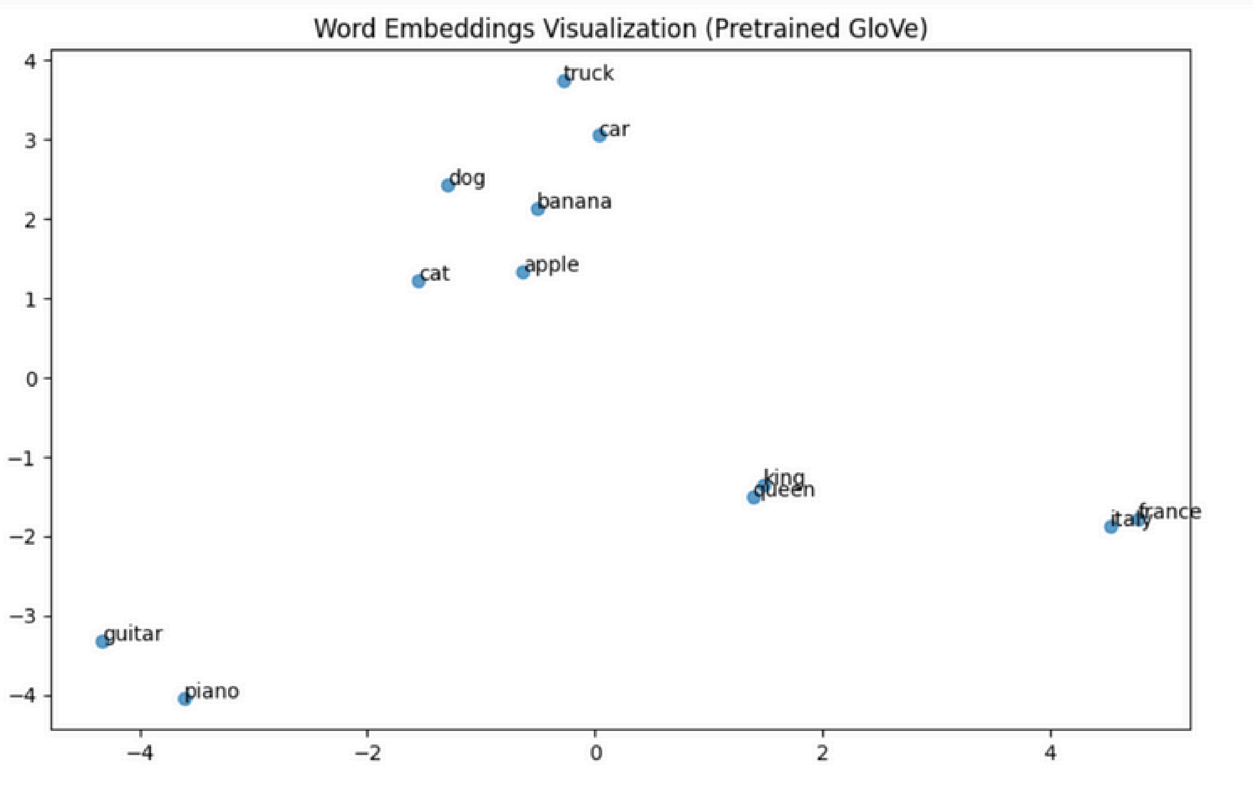
Cosine Similarity Before and After Alignment

Word	Before	After
During	0.1296	0.2576
This	-0.0047	0.6286
Money	0.0388	0.3373

From this table we can find that the similarity between words has increased significantly after the word alignment.

CONCLUSION

This study demonstrates the effectiveness of co-occurrence-based and neural word embeddings in capturing word semantics. Cross-lingual alignment using Procrustes Analysis successfully mapped Hindi word vectors to the English space, enabling better multilingual applications.



REFERENCES

IIT Bombay Parallel Corpus

https://www.cfilt.iitb.ac.in/iitb_parallel/dataset.html

FastText Pre-trained Embeddings

https://www.cfilt.iitb.ac.in/iitb_parallel/dataset.html

Procrustes Analysis for Embedding Alignment

https://doi.org/10.48550/arXiv.1702.03859

SimLex-999

https://doi.org/10.48550/arXiv.1702.03859

UMAP

https://umap-learn.readthedocs.io/en/latest/