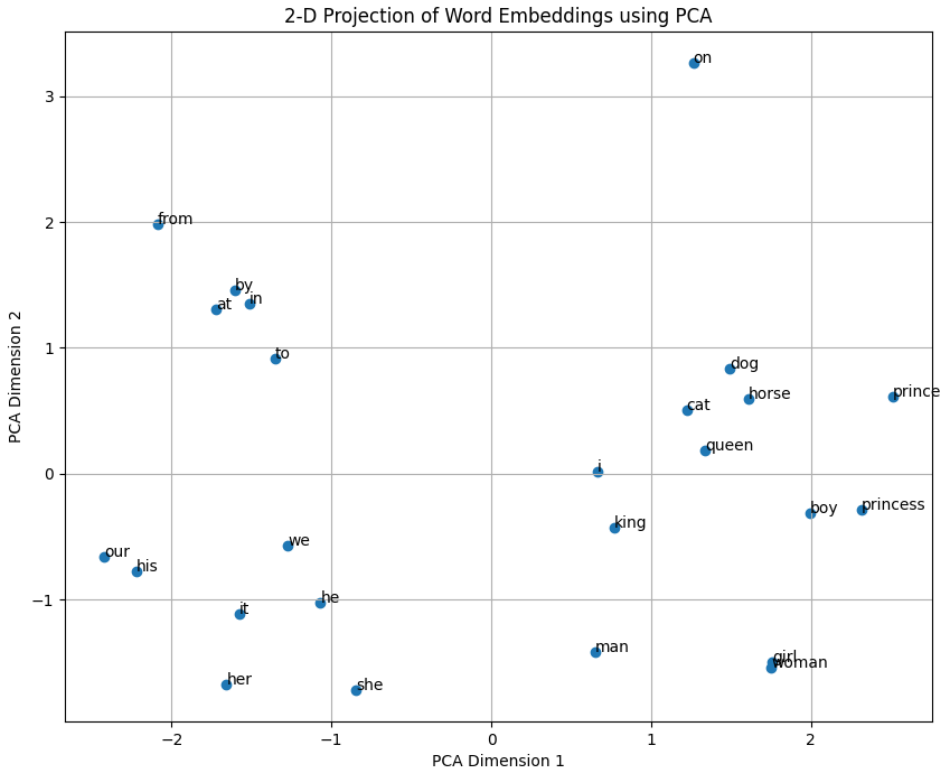
Report:

1. The best learning rate is 0.0001.
2. Cosine similarity:
    a. Closer pairs in word embedding space:
        i. [cat, tiger] = 0.73 or [plane, human] = 0.44. So, [cat, tiger] pair is closer.
        ii. [my, mine] = 0.58 or [happy, human] = 0.51. So, [my, mine] pair is closer.
        iii. [happy, cat] = 0.61 or [king, princess] = 0.62. So, [king, princess] pair is closer.
        iv. [ball, racket] 0.40 or [good, ugly] = 0.46. So, [good, ugly] pair is closer.
        v. [cat, racket] = 0.56 or [good, bad] = 0.56. So, both pair is similar.
    b. Analogy test:
        i. king:queen, man: woman
        ii. king:queen, prince: smiles
        iii. king:man, queen: woman
        iv. woman:man, princess: acadians
        v. prince:princess, man: woman
3. Cosine similarity:
    a. Closer pairs in word embedding space:
        i. [dog, cat] = 0.73 or [car, grape] = 0.52. So, [dog, cat] pair is closer.
        ii. [apple, grape] = 0.51 or [coffee, tea] = 0.70. So, [coffee, tea] pair is closer.
        iii. [guitar, apple] = 0.55 or [piano, violin] = 0.62. So, [piano, violin] pair is closer.
    b. Analogy test:
        i. earth:sun, moon: shines
        ii. book:author, song: snabb
        iii. man:he, woman: she
4. The given `eval_embs.py` file give me the following result:
   Word Similarity Test Pearson Correlation: 0.13715858934140918
   Accuracy on Analogy Test: 0.042180277349768876
5. We used deep neural network. In this paper they used a shallow linear layer model.
   We used cross entropy loss which uses softmax type activation. They used hierarchical softmax to avoid computational expence because no. of classes was high.
   We used CBoW that doesn't takes word order into account. They used bag of n-grams that captures some partial information about local word order. Actually they trained fastText with and without bigrams.
   For the tag prediction task they remove the words and tags occurring less than 100 times. We didn't do any such filtering.

Extra Credit:

1. Below is the 2-D projection of the embeddings for words in the question:



The above 2-D projection of word embeddings using PCA reveals several notable patterns: Common prepositions like 'by,' 'in,' 'to,' and 'from' cluster together, with the exception of 'on.' Pronouns 'he,' 'she,' 'his,' and 'her' share a similar direction, indicating related contextual usage. Animal names 'dog,' 'cat,' and 'horse' also cluster together, suggesting similar semantic contexts. The analogy relationship between royalty terms 'king,' 'queen,' 'prince,' and 'princess' is evident from their aligned directions. In contrast, words like 'boy,' 'girl,' 'man,' and 'woman' do not exhibit clear directional meaning, implying more complex relationships in the embedding space.

Using a 2-dimensional projection of word embeddings can be misleading for several reasons:

Loss of Information: High-dimensional embeddings contain rich syntactic and contextual information, which might get oversimplified in 2D and lose subtle nuances.
Non-Linear Relationships: Word relationships are often non-linear, and a 2D projection typically assumes linear relationships between points. As a result, non-linear relationships may not be accurately represented in a 2D projection, leading to misinterpretations of word associations.
Overlap: In a 2D projection, words that are semantically distinct in higher dimensions may overlap or appear close to each other.

So, 2D projections can provide visualization of word embeddings, but they should be used with caution, especially when making inferences about semantic or syntactic relationships between words.