**Experimental Setup**

**The Dataset**

For training our word-embedding models we used the Roman-Urdu Parl dataset [Refer]. This is a parallel corpus in Urdu and Roman Urdu Languages. [Refer] mentions that the dataset had been collected from 22 different web sources and covers text from a wide range of subjects. The dataset had been extensively preprocessed to bring parallelism to it. Furthermore, the Roman Urdu dataset also caters to Roman Urdu’s lack of standardization by containing same words spelled differently, this makes the dataset diverse and suitable for training.

The datasets used had to be pre-processed before we proceeded with feeding them to our models for the training process. For both datasets we had to remove numbers, punctuation marks and had to replace multiple whitespace characters with spaces as these would interfere with the quality of the learned embeddings. For the Roman Urdu corpus, we had to convert all words to lower case in order to avoid similarly spelled words being assigned different word embeddings by our model. This was done so to keep a level of consistency with the Urdu models as the concept of capitalization is foreign to the Urdu Language. The word and sentence count of each dataset after preprocessing is described in the table below:

|  |  |  |
| --- | --- | --- |
|  | **Roman Urdu** | **Urdu** |
| **Unique Words** | 37,172 | 38,144 |
| **Total Words** | 92,217,849 | 92,314,560 |
| **Total Sentences** | 6,365,474 | 6,365,509 |

Although both the datasets are parallel the slight differences in their counts can be attributed to the lack of standardization exhibited by the Roman Urdu language as explained in [refer]

**Model Training Parameters**

We trained each word embedding model over both corpora separately. For each model we kept the training parameters same to remove possible bias for our analysis phase. An embedding size of 500 was chosen for our word vectors, larger sized vectors tend to pack more information with regards to the learned embedding. The context window size was set to 5, which meant that while determining the context of a particular word the previous and next 5 words were taken into account. Min Count was set to 1, this variable takes a word into account for an embedding if it appears in the corpus at least once. This was done in order to cater to each and every rare word in the corpus. The fastText model had an additional parameter ‘n\_grams’, which is the number of sub-words a word is to be broken into while learning its embedding. This was kept to 3. After training each model over the set of stated parameters, we saved the learned embeddings for the purpose of our evaluation and analysis.

**Evaluation**

**Quantitative Analysis using Spearman’s Rank Correlation:**

For a quantitative evaluation of our models on these datasets, we used Spearman’s Rank Correlation which is a technique used to calculate the strength and direction between two variables. It returns a value from -1 to 1 where:

* +1: a perfect positive correlation between variables
* -1: a perfect negative correlation between variables
* 0: no correlation between variables

The Spearman’s Rank Correlation is represented by the formula:

ρ = 1-

where,

* + n is the number of observations,
  + di is the difference in rank between the ith observations

// add how variable analysis done here in spearman’s rank

**Qualitative Analysis:**

Explain basis for our qualitative analysis