A dataset to evaluate Hindi Word Embeddings

Vimal Kumar Soni†, Dinesh Gopalani‡ and M C Govil§
Department of Computer Science and Engineering
Malaviya National Institute of Technology, Jaipur, India
E-mail: †ruvimals@gmail.com, ‡dgopalani.cse@mnit.ac.in, §govilmc@gmail.com

Abstract. The current trend to solve different challenges of Natural Language Processing utilizes various online crawling methods to fetch the data and applying different shallow or deep learning methods to develop models for the respective tasks on this data. Word vectors generated using such methods are being applied for several NLP challenges and such vectors are being evaluated on word similarity task. Not only huge data is available but also multiple datasets are available for the English language to evaluate the performance of the developed models. However, the scenario is not the same for Indian languages specifically for Hindi. Focusing this challenge, we propose a dataset to check word similarity in Hindi. The construction process and afterwards annotation process are described in details. To construct this dataset, first, 353 word-pairs from the most popular English dataset are selected and translated. Their translations are verified by Hindi Experts. These word pairs are finally annotated independently by 11 native Hindi speakers. Multiple criteria have been set to select the annotators for this task. The final dataset has been evaluated on CBOW and Skip-gram models.

1. Introduction
The Natural Language Processing deals with text written in different scripts and there is a huge gap in the pace of research among various languages. Enormous data from different domains as well as several tools are available for the English Language. The reason behind this is that English has been associated with computers since starting. The Internationalization process helped English to reach to the outskirts in a bundled pack with computers. However, for other languages, the situation is completely different as most of the computers have default language as English. In the past, computers used to have Indian language characters as non-standard characters however the scenario changed with the availability of language interface packs for Indian Languages and standardization of characters of scripts for Indian Languages. After the birth of Social Networking in 2003, the explosion of information happened and raw data of various languages started appearing on the internet and grew rapidly. This resulted in the availability of huge raw data which caters the need of bulk data for unsupervised learning. With the advancement in neural networks in the form of shallow and deep learning, such crawled data are being used for unsupervised training to learn the semantic relations among the vocabulary words. These semantic relations are represented in the form of word vectors which can be used for various NLP challenges [1]. To evaluate such word vectors, word similarity task is quite popular and has already been accepted as a benchmark. Word vectors are applied to ambiguity resolution [2], automatic machine translation [3], next-word prediction [4], opinion mining [5] and question generation & answering [6]. The quality of word-vectors plays an important role in such applications as these vectors represent the semantics of the word in the form of vector. Therefore, the evaluation of such vectors is quite essential. Word similarity task has been proved to evaluate such vectors efficiently. A dataset which contains human-annotated semantic similarity index between word-pairs is used as a standard reference. Based
on this reference, between the semantic similarity index and the respective word vectors, a correlation is calculated. Several research works are in progress to take leverage of word embeddings for the English language and different datasets are also available to evaluate these word embeddings but not for Hindi language. Though Hindi is the fourth most spoken language in the world yet the number of available computational resources for this language is low. A number of attempts are in progress to generate word embeddings for the Hindi language which can be used to address various NLP challenges, though the unavailability of a dataset to evaluate those word vectors is still a big challenge. This challenge motivated us to create a word similarity dataset to evaluate the Hindi language word vectors. The main contribution of this paper is to develop and provide the word similarity dataset for the Hindi language which is verified at multiple levels by Hindi language teachers and validated by inter-annotator agreement.

This paper is organized as follows: Initially, a summary of work in this area is presented in the next section. The process of construction of dataset is mentioned in Section-III. Section-IV describes the models and tuning parameters used to evaluate the dataset. The section-V explains about the evaluation and the section-VI discusses the results. The last section presents open research gaps.

2. Related Work

The English language is considered as a rich language in terms of availability of corpus and tools related to basic NLP tasks. In the same way, multiple datasets are available for the evaluation of word vectors. RG-65 [7] is a classical dataset which was released in 1965 and has motivated researchers of other languages to create word similarity datasets. In the year 1991, MC-30 [8] was released to show an inverse linear relationship between similarity of meaning and the discriminability of contexts. Released in 2001, WordSim-353 [9] is considered as state of the art and one of the foremost datasets for benchmarking to evaluate the word representation vectors. Yang et al explored the idea of verb similarity in 2006 [10]. Szumlanski proposed a new dataset Rel-122 based on relatedness in 2013 [11]. Li et al extended the idea of word similarity to multiword similarity and presented MWE300 in 2013 [12]. In 2015, Hill et al. used different POS categories to develop Simlex-999 dataset [13]. Gerz et al further explored the categories and developed a dataset for verb similarity in 2016 [14]. Different versions of WordSim-353 for other language are also developed. Netisopakul has used WordSim-353 and SimLex-999 to create Datasets for Thai Language [15]. Reference [16] developed multiple datasets for Indian languages namely Urdu, Marathi and Telugu using WordSim-353 and RG-65. There are various ways to generate word embeddings [17], however, all of these can be evaluated by word similarity task.

3. Dataset Construction

The steps involved in the construction of this dataset are shown in Figure-1.
3.1. Preparation of Data
Several datasets are available in the English language for word similarity task. To construct one for the Hindi language, we have taken the most popular word-similarity dataset from English which is WordSim-353. It contains 353 word-pairs which are domain-independent. This dataset has also been used to create datasets for other languages.

3.2. Translation
These word-pairs are then translated by an expert of Hindi language. The expert has been instructed to remain neutral for the translation of the individual word and not to take the context of any word from its pair. Few examples of Context-Free Vs Context-Based Translation from the word-sim353 are mentioned in the Table-1. Since the translation was based on the individual word so Context-Free Translation is preferred.

| Context-free vs context-based translation |
|-------------------------------------------|
| **Word-1 Hindi Translation** | **Word-2 Hindi Translation** | **Word-Pair Hindi Translation** |
| drug | दवा | गाली | नशाखोरी |
| cell | कोशिका | फोन | सैल्फोन/ मोबाइल-फोन |

There are words which are not part of Hindi Language, yet, they have become commonly accepted due to heavy usage, such words have been transliterated as it is instead of their exact translation. This has been done due to the fact that the exact translated word would have rare occurrence but the transliterated form has more occurrences. Few examples of such translation and transliteration cases are mentioned in Table-2 along with the accepted form used for the dataset.

| Translation vs Transliteration |
|-------------------------------|
| **Word** | **Hindi Transliteration** | **Accepted Form** |
| computer | संगणक | कम्प्यूटर |
| report | रपट | रिपोर्ट |
| theater | नाट्यशाला | थिएटर |
| film | चलचित्र | फिल्म |

The specific names and acronyms have been accepted with their widespread transliterated versions in this dataset as shown in Table-3

| Transliteration of Proper Nouns |
|-------------------------------|
| **Word** | **Hindi Transliteration** | **Accepted Form** |
| Arafat | अराफात | अराफात |
| Dollar | डॉलर | डॉलर |
| Jerusalem | यरूशलेम | यरूशलेम |
| OPEC | ओपीईसी | ओपीईसी |

However, for the sake of consistency, only one meaning of a word has been taken irrespective of its occurrences in either side of word pair. Table-4 shows few words with multiple occurrences. The number of occurrences as the first part of pair and the last part of pair in the original wordsim353 dataset along with their most common Hindi Translation forms which are accepted for our dataset are also mentioned.
The outcome of this step is a domain-independent set of 353 word-pairs in the Hindi Language.

### 3.3. Selection of Annotators

After finalizing the word-pairs in the Hindi Language, these are given to annotators to fill the similarity score manually for each word pair. There are three conditions kept to select an annotator for this task. First, Hindi must be the mother-tongue of the annotator and Second, he or she must have studied the Hindi language at least up to Senior Secondary Level and third, that he or she must have completed the graduation in any subject. Most of the annotators are either primary school teachers or postgraduate scholars and are directly related to the education domain. Total of 11 annotators provided this score.

### 3.4. Annotation and Scoring Process

The process of annotation is explained to the annotators. Annotators are asked to give a score for similarity to a word pair in a range of 0 to 10. Here 0 is to be assigned to totally non-similar word-pair while 10 is to be assigned to if both words in the word-pair are identical or have the perfectly same meaning. Annotators are also asked to fill the similarity of the words on their assessment instead of a mutual discussion with others to finalize the score. The experts involved in the translation process are not given the role of annotator to keep the annotation score free from any kind of bias which could occur due to the familiarity of word-pair during the translation process.

### 3.5. Post-processing & Finalizing the Similarity Score

To calculate the similarity score for each word-pair, the similarity scores provided by all annotators for the newly developed dataset are averaged over the collected 11 responses for each word-pair. The average score is taken as the final similarity score for a word-pair.

### 4. Model and Tuning Parameters

To evaluate the word vectors, first, such word embeddings are generated & trained using the following Corpus, Learning Models and Tuning parameters-

#### 4.1. Corpus

The Monolingual corpus developed by IIT-Bombay is used to train the model [18]. This corpus contains approximately 45 million sentences. The newly developed dataset is used for testing purpose. A zero vector is assigned to Out-Of-Vocabulary (OOV) words. To ensure canonical Unicode representation, this monolingual corpus is normalized using the Indic NLP Library.

#### 4.2. Learning Models

To learn the word embedding, the word2vec tool with CBOW & Skip-gram model is used. Along with these, the fastText Model is also used to generate the Hindi word vectors.

#### 4.3. Tuning Parameters

As far as the tuning parameters are concerned, the dimension of the vectors is set to 300. The negative sampling is applied and set to 5 with a window of 5 words. The minimal number of word occurrences is also set to 5 to skip the rarely occurred words.

---

**Table 4. Frequency of Occurrence**

| Word       | Number of occurrences as the first part of pair | Number of occurrences as the last part of pair | Most Common Hindi Translation |
|------------|-----------------------------------------------|-----------------------------------------------|-------------------------------|
| money      | 12                                            | 2                                             | धन                           |
| psychology | 12                                            | 0                                             | मनोविज्ञान                     |
| stock      | 7                                             | 2                                             | शेयर                          |
| tiger      | 10                                            | 1                                             | बाघ                           |
5. Evaluation

5.1. Inter Annotator Agreement
The semantics of a word not only changes according to context but also different readers can interpret that word in different ways. The impact of subjectivity can influence the annotation process. Therefore to check the impartiality and neutrality of the annotation process, an inter-annotator agreement is measured. This agreement defines the sanctity of the annotation along with the preciseness in the interpretation of one word with respect to the other word.

To calculate the inter-annotator agreement, several statistical measures are available like Cohen’s kappa and Fleiss’ kappa. Cohen’s kappa is used to calculate the inter-annotator agreement between two annotators only whereas Fleiss’ kappa is used as a measure to calculate the inter-annotator agreement for more than two annotators. For this dataset, the Fleiss’ Kappa score for inter-annotator agreement is 0.79 which shows a good agreement among all annotators.

| Table 5. Fleiss’ Kappa Score |
|-----------------------------|
| Dataset | Number of Annotators | Fleiss’ Kappa score |
| Hindi | 11 | 0.79 |

6. Result and Analysis
This word similarity dataset is evaluated on the state of the art models – Word2vec (with CBOW & Skipgram) and fastText as per the tuning parameters mentioned in section 4. The opted baseline scores are mentioned in the Table-6.

| Table 6. Baseline Scores |
|--------------------------|
| System | Training Corpus Size | Score |
| Word2vec-CBOW | 45 million Sentences | 24.97 |
| Word2vec-SkipGram | 45 million Sentences | 27.80 |
| fastText | 45 million Sentences | 29.11 |

These trained models come across some words that do not appear during training (out of vocabulary words). It can be observed from Table 6 that fastText outperformed word2vec system. A random subset of the dataset is mentioned in the Table-7.

| Table 7. Excerpts of HindiWordSim-353 |
|--------------------------------------|
| S. No. | Word Pair | Similarity Score |
|--------|-----------|------------------|
| 1      | बाघ बाघ   | 10               |
| 2      | बाघ तेंदुआ | 7.73             |
| 3      | तेंदुआ बालक | 9.36           |
| 4      | टेलीफोन संचार | 5.55          |
| 5      | महीना होटल | 2.36             |
| 6      | राजा पत्तागोभी | 1.64          |
| 7      | फुटबॉल फुटबॉल | 10             |
| 8      | बैंक धन    | 8.64             |
| 9      | वित्तस्थर अलमारी | 4.09          |
| 10     | टेलीविजन रेडियो | 6.73          |
| 11     | सदी राष्ट्र    | 3.18             |

The complete dataset is made available at https://github.com/ruvimals/Dataset-Hindi-WordSim353
7. Conclusion & Future Work

Multiple research works are being carried out using word vectors for natural languages. Resource-rich languages such as English or French are getting highly benefitted. However, widely spoken Hindi language is lacking due to scarcity of resources. Now with the current trend, many resources are being created and added to the resource-list of Indian languages. This work is a contribution towards the creation of resources for the Hindi language.

In future, this research work can be continued to develop the dataset with a ratio of POS categories. Another challenge is to create a domain-specific dataset to carry out the research work in domains like agriculture or tourism.

References

[1] Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).
[2] Hadiwinoto, Christian, Hwee Tou Ng, and Wee Chung Gan. "Improved word sense disambiguation using pre-trained contextualized word representations." arXiv preprint arXiv:1910.00194 (2019).
[3] Choi, Heeyoul, Kyunghyun Cho, and Yoshua Bengio. "Context-dependent word representation for neural machine translation." Computer Speech & Language 45 (2017): 149-160.
[4] Tenney, Ian, et al. "What do you learn from context? probing for sentence structure in contextualized word representations." arXiv preprint arXiv:1905.06316 (2019).
[5] Rana, Ankur, Vishal Goyal, and Vimal K. Soni. "Renaissance of Opinion Mining." International Conference on Information Systems for Indian Languages. Springer, Berlin, Heidelberg, 2011.
[6] Medved, Marek, and Ales Horák. "Sentence and Word Embedding Employed in Open Question-Answering." ICAART (2). 2018.
[7] Rubenstein, Herbert, and John B. Goodenough. "Contextual correlates of synonymy." Communications of the ACM 8.10 (1965): 627-633.
[8] Miller, George A., and Walter G. Charles. "Contextual correlates of semantic similarity." Language and cognitive processes 6.1 (1991): 1-28.
[9] Finkelstein, Lev, et al. "Placing search in context: The concept revisited." Proceedings of the 10th international conference on World Wide Web. 2001.
[10] Mahrishi, M., Hiran, K. K., Meena, G., & Sharma, P. (2020). Machine Learning and Deep Learning in Real-Time Applications. IGI Global.
[11] Szumlanński, Sean, Fernando Gomez, and Valerie K. Sims. "A new set of norms for semantic relatedness measures." In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 890-895. 2013.
[12] Li, Peipei, Haixun Wang, Kenny Q. Zhu, Zhongyuan Wang, and Xindong Wu. "Computing term similarity by large probabilistic isa knowledge." In Proceedings of the 22nd ACM international conference on Information & Knowledge Management, pp. 1401-1410. 2013.
[13] Hill, Felix, Roi Reichart, and Anna Korhonen. "Simlex-999: Evaluating semantic models with (genuine) similarity estimation." Computational Linguistics 41, no. 4 (2015): 665-695.
[14] Gerz, Daniela, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. "Simverb-3500: A large-scale evaluation set of verb similarity." arXiv preprint arXiv:1608.00869 (2016).
[15] P. Netisopakul, G. Wohlgenannt and A. Pulich, "Word Similarity Datasets for Thai: Construction and Evaluation," in IEEE Access, vol. 7, pp. 142907-142915, 2019, doi: 10.1109/ACCESS.2019.2944151.
[16] Akhtar, Syed Sarfaraz, et al. "Word similarity datasets for indian languages: Annotation and baseline systems." Proceedings of the 11th Linguistic Annotation Workshop. 2017.
[17] Soni, Vimal Kumar, Dinesh Gopalani, and M. C. Govil. "Resurgence of Deep Learning: Genesis of Word Embedding." Smart Innovations in Communication and Computational Sciences. Springer, Singapore, 2019. 129-139.
[18] Anoop Kunchukuttan, Pratik Mehta, Pushpak Bhattacharyya. “The IIT Bombay English-Hindi Parallel Corpus.” Language Resources and Evaluation Conference. 2018.