Linguistic features for Hindi light verb construction identification

Ashwini Vaidya
IIT Delhi
ird11278@ee.iitd.ac.in

Sumeet Agarwal
IIT Delhi
sumeet@iitd.ac.in

Martha Palmer
University of Colorado, Boulder
martha.palmer@colorado.edu

Abstract

Light verb constructions (LVC) in Hindi are highly productive. If we can distinguish a case such as *nirnay lenaa* 'decision take; decide' from an ordinary verb-argument combination *kaagaz lenaa* 'paper take; take (a) paper', it has been shown to aid NLP applications such as parsing (Begum et al., 2011) and machine translation (Pal et al., 2011). In this paper, we propose an LVC identification system using language specific features for Hindi which shows an improvement over previous work (Begum et al., 2011). To build our system, we carry out a linguistic analysis of Hindi LVCs using Hindi Treebank annotations and propose two new features that are aimed at capturing the diversity of Hindi LVCs in the corpus. We find that our model performs robustly across a diverse range of LVCs and our results underscore the importance of semantic features, which is in keeping with the findings for English. Our error analysis also demonstrates that our classifier can be used to further refine LVC annotations in the Hindi Treebank and make them more consistent across the board.

1 Introduction

Light verb constructions (LVC) are found across languages e.g. Japanese, Korean, Persian as well as English. An LVC consists of a predicating element (usually a noun) and a verb, which is also known as a *light* verb. For instance, *take a walk* or *give a sigh* are LVCs consisting of light verbs *take* and *give* and their corresponding predicating nouns *walk* and *sigh*. The nouns in an LVC contribute to the event semantics and the light verb supplies additional meaning e.g. agentivity, completeness, or permission. In Hindi, LVCs are productive and are also sometimes termed as 'support verb' or 'conjunct verb' constructions. Examples 1 and 2 contrast the use of a simple predicate *de* 'give' with its light verb usage.

(1) Simple predicate

raam=ne mohan=ko kit̪ab d̪-ii
Ram.M.Sg=Erg Mohan.M.Sg=Dat book.F.Sg give-Perf.F.Sg

'Ram gave Mohan a book'

(2) Noun-Verb complex predicate

raam=ne us baat̪=par **zor** **d̪i-yaa**
Ram.M.Sg=Erg that topic=loc pressure.M.Sg give-Perf.M.Sg

'Ram put an emphasis on that topic'

LVCs form a large part of the lexicon in Hindi. In the Hindi treebank (Palmer et al., 2009) (400,000 words), there are nearly 47,163 predicates, of which 37% have been annotated as LVCs. LVCs consist of predicate types that are far more numerous than simple verbs. Hindi has approximately 700 simple verbs, but potentially many more unique LVCs. This makes them a highly productive phenomenon in Hindi.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1320–1329, Osaka, Japan, December 11-17 2016.

Butt (2010) notes that light verbs in Hindi LVCs act as a verbalizers in order to create new predicates and incorporate borrowed items into the language e.g. *email kar* 'email do; email'. Therefore, LVCs are sometimes described as "a preferred way of augmenting the creative potential of the language" (Kachru, 2006, pp 93).

The identification of LVCs in Hindi (as well as other South Asian languages) is an important NLP task, which has been shown to improve parsing accuracy (Begum et al., 2011) as well as machine translation performance (Pal et al., 2011). The detection of multi-words such as LVCs has been widely studied and association measures, linguistic knowledge and parallel corpora have been used.

As LVCs are a type of multiword, a commonly used method is 'N-gram classification' (Green et al., 2013). This strategy extracts n-grams from the corpus, filters them and assigns some values based on a bigram measures such as log-likelihood or mutual information. A classifier is then used to make a LVC/non-LVC decision. However, previous work has shown that LVCs benefit from the use of linguistic features for identification. Vincze et al. (2011) used bigram association measures for English noun-noun compounds and LVCs. They found that LVC detection improves when linguistic features are used in addition to n-gram information. Tu and Roth (2011) showed that linguistic and statistical features perform at par for English LVC detection.

For Hindi, we expect that linguistic features will be useful for automatic detection. At the same time, the productivity and range of LVC constructions in Hindi result in some specific challenges. In the next section, we carry out a linguistic analysis of LVCs based on the annotations in the Hindi Treebank. We use these insights to propose two new features to identify LVCs. Following this, we describe our experimental setup and then discuss the results.

2 Linguistic challenges for Hindi

The linguistic notion of an LVC differs across languages. While annotating an English corpus with LVC information, Tu and Roth (2011) make use of a 'replacing' principle for their annotators, where if a candidate light verb like *take* in *take a walk* can be replaced by *walk* without (too much) of a change in meaning, then a combination like *take a walk* is considered an LVC.

In Hindi, such a 'replacing' principle is not available as the nouns that participate in LVCs are not necessarily deverbal in nature i.e. the majority do not have a direct verbal counterpart. In fact, LVCs are a preferred method of introducing new predicates into the language via borrowed nouns. Bhattacharyya et al. (2007) have described a number of diagnostic criteria for Hindi LVCs, but these are not completely robust–and can only be applied to LVCs that are transitive. In fact, most linguistic diagnostics mentioned in Mohanan (1994) and Bhattacharyya et al. (2007) are appropriate for transitive LVCs that occur with light verb *kar*. Such cases are the most frequently occurring LVCs in Hindi, but do not represent all LVCs.

Consequently, the application of diagnostic tests for LVCs for a large corpus can be challenging. The Hindi Treebank (Palmer et al., 2009) is a relatively large resource that is annotated with LVC information using the `pof` label. We use this data to examine the behaviour of Hindi LVCs, focusing on each component: the light verb and the predicating nominal.

2.1 Light verbs

Jespersen (1965) coined the term 'light' verb to refer to verbs that do not behave like standard verbal predicates as they have a depleted semantic contribution to the event described by the LVC. These light verbs tend to be similar cross-linguistically e.g. *take*, *make* and *give* can be found in English, Persian and Hindi. At the same time, these verbs are distributed differently across languages.

In the Hindi Treebank, the distribution of light verbs reflects some interesting facts about LVC formation in Hindi. Figure 1 shows the 20 most frequently occurring light verbs in the Training and Development sections of the Hindi Treebank (approx. 21,000 sentences). These light verbs include the following: *kar* 'do', *ho* 'be/happen', *de* 'give', *hE* 'be', *raha* 'stay', *aa* 'come', *karaa/karvaa* 'cause to do', *lagaa* 'touch/feel', *jataa* 'convey', *le* 'take', *banaa* 'make', *rakh* 'keep', *chal* 'go', *uthaa* 'rise', *daala* 'put', *laDa* 'fight', *lag* 'seem', *ban* 'become', *maar* 'hit'. Each of these light verbs also appear as 'full' verbs

i.e. they can also appear without a nominal predicate, as a non-LVC.

The bar plot in Figure 1 shows that the frequency of *kar* 'do' is the greatest, followed by *ho*, 'be' and *de* 'give'. The light verb *kar* 'do' has many more positive cases of LVCs as compared to non-LVCs.For other light verbs such as *de*, the distribution is more even and with other light verbs, there are far more non-light usages of these verbs as compared to light.

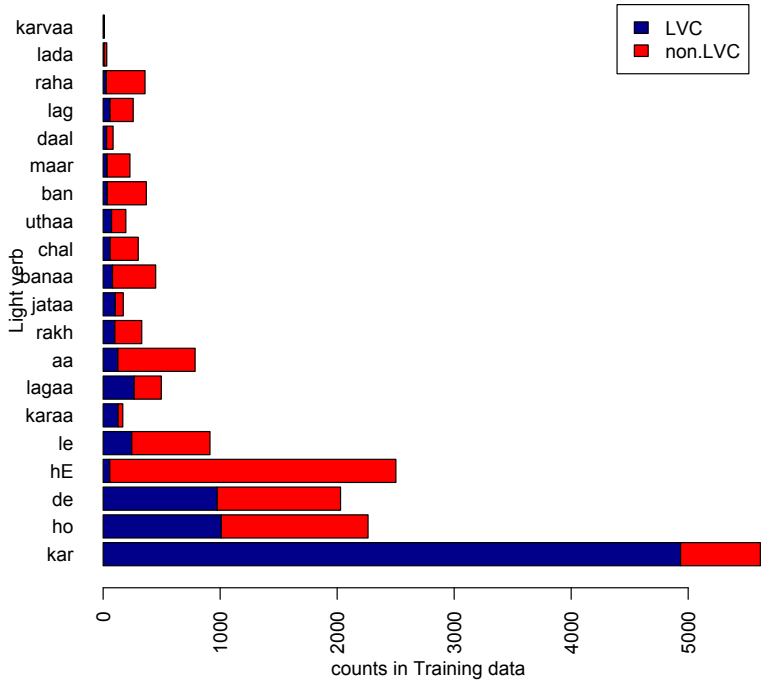

Figure 1: Light verb distribution in the Training and Development section of the Treebank

If we were to divide the training data by light verb and use the majority class to predict the light or non-light case, we would still get reasonably good results. This is because the light *kar* cases far outnumber the others, and one can expect the majority class baseline to be as high as 0.8. Conversely, a light verb like *aa* 'come' has more non-light usages than light, hence the majority class prediction would also be quite high.

Begum et al. (2011) describe a classifier for Hindi LVCs but do not mention the distribution of LVCs in the data. Therefore, it is difficult to know whether the results are applicable to all LVCs or just the light verb 'kar'. In contrast, Butt et al. (2012) focus on light verbs *kar* 'do' and *ho* 'be' alone. In this paper, we make use of the Hindi Treebank LVC annotations to evaluate our 'combined' system, but provide evaluation across individual light verbs in the corpus. This also implies that we must incorporate features that are specific not only to 'kar', but across all light verbs in the data.

2.2 Nominal predicates

The Hindi Treebank consists of more than 3000 unique nominals that can occur as part of an LVC. At the same time, some of these nouns can combine with more than one light verb to form an LVC e.g. *ishaara kar* 'signal do; make a sign' and *ishaara de* 'signal give; give a sign', with some subtle differences in meaning. It is possible that one of these combinations is more frequent than the other- or they may be equiprobable. There are also certain nouns where only one light verb is possible e.g. *maut ho* 'death be; die'.

We carried out a corpus study, examining 1853 unique nouns from the Training and Development sections of the Hindi Treebank and extracted the number of light verbs that occurred with them. Although,

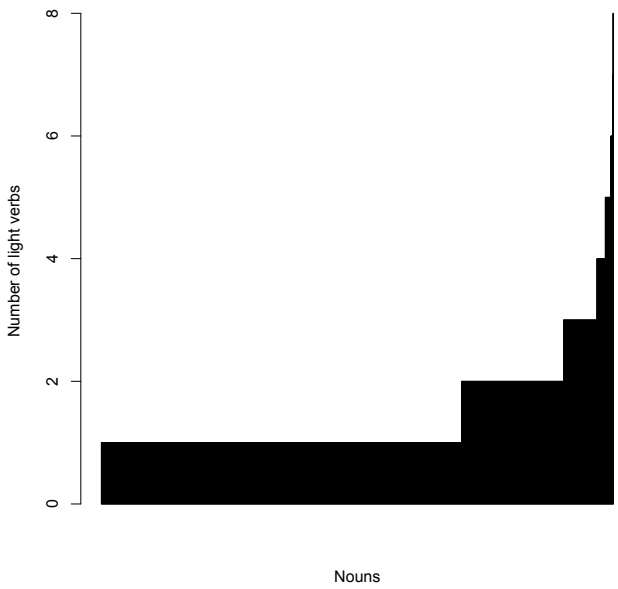

Figure 2: Number of light verbs that occur with a unique noun. (t=1853)

Figure 2 shows a 'long tail', where a large number of nouns occur with just one light verb, about 1/4th of the data consists of nouns that alternate with more than one light verb.

These alternations show that a collocational measure that only looks at the bigram occurrences may not be able to capture a noun-light verb alternation that is relatively infrequent. Therefore, linguistic information would be required to identify a predicating nominal that appears in a number of contexts. In the following section, we propose new features that could help capture this information.

3 Linguistic features used for LVCs

English LVC identification focuses on extracting linguistic features for LVCs (Tan et al., 2006; Grefenstette and Teufel, 1995; Stevenson et al., 2004). For example, the morpho-syntactic similarity between nominal predicates and their verbal counterparts (e.g. *walk* and *take a walk*) is often exploited. Other cues include the presence of indefinite determiners (such as *a*) before the nominal predicate.

Tu and Roth (2011) look at both statistical and linguistic contexts to detect English complex predicates. Among their local linguistic features, they utilize bigram information about the nominal head and light verb, the nouns themselves and the Levin verb class members of deverbal nouns. In a more recent study, Chen et al. (2015) have described an improvement over Tu and Roth (2011)'s performance by using lexical features from WordNet, as well as word sense information. Using the Tu and Roth (2011) testset, they report a 0.89 F-score for English LVCs.

Author/Feature	**Tan et al'06** (Eng)	**Tu and Roth'11** (Eng)	**Begum et al'11** (Hin)
Deverbal noun	Y	Y	
Noun semantics	Y	Y	Y
Light verb list	Y	Y	Y
Presence post-posn			Y
Presence determiner	Y	Y	
Collocational measure		Y	Y

Table 1: Commonly used linguistic features for English and Hindi LVC detection.

The work for Hindi LVC detection makes use of a similar set of linguistic features. Begum et al. (2011) look for the presence of postpositions and demonstratives, which preferentially do not occur with a noun that is a part of an LVC. Like Tu and Roth (2011), they use the verb-object bigram and the noun class information from Hindi WordNet (Narayan et al., 2002). They have achieved an accuracy of around 0.85 for identification of Hindi complex predicates. More recently, Singh et al. (2015) have compared word embeddings and WordNet-based measures to detect Hindi noun compounds and LVCs. While word embeddings are effective for compounds, they perform poorly for LVCs, suggesting the importance of more precise linguistic features.

3.1 Linguistic features for Hindi

Our linguistic analysis of LVCs indicates that the properties of both noun and light verb are crucial as features for identifying LVCs. In the previous section, we described some of the commonly used features for LVC identification. Table 1 shows some of these: the presence of post-positions after the predicating noun, collocational features and lexical features.

In this section, we introduce two new features that are based on our study of Hindi LVCs. The first is based on the idea that there are semantic constraints on the combination of a particular noun and light verb. Sulger and Vaidya (2014) examined the combinatorial properties of noun and light verb based on relative frequency of occurrence. They found that a light verb such as *de* 'give' is likelier to combine with nouns that have a 'transfer' property, whereas nouns that occur with *kar* 'do' will occur with nouns that describe actions with animate agents. Light verb *ho* 'be' often appears with stative nouns or those that indicate mental states.

In order to capture these properties, we used a feature that associated a light verb with the ontological property of the noun that is likely to occur with it. For example *kar* was associated with *Physical_Action_Abstract_Inanimate* and *de* 'give' with *Communication_Action_Abstract_Inanimate*. These ontological properties were extracted from Hindi WordNet (Bhattacharyya, 2010). If a noun occurred with the ontological property that was associated with a particular light verb, it was marked positively for this feature.

The second feature was based on the idea that predicating nominals usually introduce arguments of their own. These usually occur with the postpositions *par* 'on', *se* 'with', *ko* 'to' or *kii* 'of'. These indicate that a nominal has introduced an argument of its own–and is likely to be a predicating nominal rather than an ordinary argument of the verb. Examples 3-5 illustrate the cases where a nominal introduced an argument with *par*, *se* or *kii*.

(3) pulis=ne **logon=par** hamlaa ki-yaa
police=Erg people=loc attack.M.Sg do-Perf.M.Sg
'(The) Police attacked the people'

(4) samir=ne **mohan=se** nafrat k-ii
samir.M.Sg=Erg Mohan.M.Sg=instr hatred.F do-perf.F.Sg
'Samir hated Mohan'

(5) samir=ne **ghadii=kii** chorii k-ii
Samir.M.Sg-Erg watch.F.Sg-gen theft.F do-Perf.F
'Samir stole the watch'

This feature was introduced to overcome some of the shortcomings of the 'presence of post-position' feature on the noun (Table 1). The post-position only looks at the presence or absence of post-positions on the predicating noun, whereas the proposed feature looks at the postpositions on the nominal's *arguments*. The former feature is restricted to agentive nominals, whereas this feature is applicable to all predicating nominals that license arguments.

4 Experimental setup

We make use of the Hindi Treebank data to train our LVC identification system. The Hindi Treebank annotation guidelines describe the use of the label `pof` to identify Hindi LVCs. They make use of

	Train	**Development**	**Test** (Treebank)	**Test** (ICON)
News	14282	3500	1708	2757
LVC	6739	1665	790	1056
non-LVC	7543	1835	918	1701

Table 2: Training, Development and Test instances, with the number of light and non-light verbs

annotators' linguistic intuition to identify **pof** cases with a "full understanding that it may lead to some inconsistency in the data" (Bharati et al., 2012, p41). This is probably because of the lack of reliable linguistic diagnostics mentioned earlier in section 2. As a result, we can think of the Hindi Treebank annotation of LVCs as reflecting a fairly generous conceptualization of LVCs.

In order to reduce any errors and inconsistencies, in this work we take into consideration only the top 20 most frequently occurring light verbs in the corpus[1]. These account for 90% of the LVCs in the Treebank. The remaining light verbs occur only less than 10 times in the corpus and we assume that low frequency might indicate an annotation error. Although this may leave out some valid cases of LVCs, we make the assumption that LVCs represented by these 20 light verbs give us a fairly good representation of the LVC construction. Begum et al. (2011) also make use of the 20 most frequently occurring light verbs in their model. However, they do not provide a list of these light verbs in their paper. Although we may not be able to make a very exact comparison with their model, we would imagine that the differences will be minor, with respect to the low frequency light verbs.

In order to select candidates, we identified positive and negative instances of LVCs in the Hindi parse trees. In the Hindi dependency parse tree, the predicative noun is a dependent of the light verb and in the majority of the cases, both noun and light verb occur next to each other in the sentence. The noun and light verb can be scrambled away from each other, but we found this to be fairly rare in the Treebank LVC examples. Therefore, we chose candidates based on proximity; e.g. if a phrase annotated as an NP occurred next to a verb phrase containing a light verb, this was taken to be a candidate for LVC identification. Apart from NP phrases, we also accepted phrases annotated as 'BLK', which indicated that the noun was borrowed from English. Such nouns often occur as part of LVCs, as complex predication is used to introduce new words into the language.

Our training data made use of the splits provided by the Hindi Treebank, to which we added a small sub-part of the Treebank consisting of conversational data, taken from fiction. The rest of the Hindi Treebank is news text. We made use of the training and test splits given by the Treebank, but kept a small portion of the training set as a development set. Table 2 shows the division between the training, development and test sets and the distribution of positive and negative classes. Across the board, we find that the number of non-LVCs is higher than the LVC instances.

The two test sets are drawn from different genres. The Treebank test set is from the testing split provided by the Hindi Treebank and is news text. The second test set consists of sentences taken from literary criticism. This data is not from the Treebank, but taken from the ICON 2009 Shared task for Hindi parsing (Husain, 2009) and we will refer to this test set as 'ICON'. We included this test set to compare the performance of our model with Begum et al. (2011).

4.1 Features

The features used for identification of LVCs can be grouped into roughly four categories viz. lexical, morphosyntactic, collocational and semantic. We used features that are similar to those included in Begum et al. (2011) as well as Tu and Roth (2011), and additionally introduced two new features (section 3.1). Table 3 shows the set of features used for identifying LVC cases in the Treebank. The other features have been used in previous work to identify English or Hindi LVCs. For the collocational features, the values were obtained from a large corpus (Hindi Wikipedia) and then converted to binary features. For log-likelihood, this was done using a table of critical values to decide whether the ratio was significant.

[1]These verbs are listed in section 2.1

Accordingly, it got the binary feature 0 or 1. In the case of PMI, we checked whether its value was greater than or less than 0 for a given noun and verb candidate. If it was greater, then the noun-verb pair was likely to be a better collocation.

Type	No	Feature
Lexical	1	Verb lemma (Baseline)
	2	Noun lemma
Morpho-syntactic	3	Postposition after noun
	4	Arguments of eventive noun (eventive nouns have an 'extra' argument)
Collocational	5	Log-likelihood value
	6	Pointwise Mutual Information value
Semantic	7	Ontological category of noun
	8	Acceptability of noun with a given light verb

Table 3: Features used for LVC detection

4.2 Model

We experimented with two types of models: a linear model (Logistic Regression) and SVM with an RBF kernel. The eight feature types described earlier generated 3278 features, over which we performed feature selection to choose 1638 (roughly half) of the features based on their individual f-scores. The motivation to carry out feature selection was because of the large number of lexical features, some of which may not have been significantly useful for the classifier. We made use of the *fselect.py* tool for feature selection (Chen and Lin, 2006).

We used the Scikit-learn package (Pedregosa et al., 2011) to train our model. Scikit-learn uses the LIBSVM implementation of support vector machines (Chang and Lin, 2011) and the LIBLINEAR implementation for logistic regression (Fan et al., 2008). We made use of 10-fold cross validation to find the best value of C and gamma for the RBF kernel (C=1, gamma=0.05).

5 Evaluation

We trained our two models using the features described in section 4.1 and evaluated them against the two test sets that we described earlier. We used the verb lemma as our baseline feature. Table 4 shows the performance of our system in comparison to the verb lemma baseline and the system described in Begum et al. (2011).

	Logistic Regression			**SVM with RBF**		
ICON	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
LVC	87.77	76.13	81.54	88.42	76.7	82.15
Non-LVC	86.31	93.41	89.72	86.63	93.76	90.06
Accuracy	86.79			**87.23**		
Begum et. al. (2011)	85.28					
Verb lemma Baseline	75.87			75.66		
News	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
LVC	86.36	91.39	88.80	85.8	90.25	87.97
Non-LVC	92.20	87.58	89.83	91.22	87.14	89.13
Accuracy	89.34			88.58		
Verb lemma Baseline	80.97			80.91		

Table 4: Precision, recall and F1 scores for the ICON and News test sets

Both our models perform better than Begum et al. (2011) on the same test set. Additionally, we also evaluated our system on the news test set from the Hindi Treebank. The performance on the news dataset is better, most probably because of the smaller number of unseen nouns in news (180) as compared to ICON (612).

The diversity of the LVCs in Hindi implies that we would like to check the performance of our system across all light verbs. As the light verb *kar* is the most frequently occurring light verb, we expect that it will give us the best results. We carried out two types of experiments for light verbs: first we

ran individual classifiers across light verbs using the same feature set and compared its micro-averaged performance with that of the combined model. We found that this result was almost exactly similar to the combined model. As a second experiment, we also looked at the performance for individual light verbs within the combined model itself.

Table 5 describes the performance for individual light verbs *kar*, *ho*, *de* and *le*. The other light verbs are fairly infrequent, hence they are grouped into *LF-TR* for transitive light verbs and *LF-INTR* for intransitive light verbs. We find that LVCs with *kar* are identified with high accuracy because of their high frequency. The performance is slightly less accurate for other light verbs, notably *ho* 'be'. However, they still perform above the baseline, indicating that the features are robust enough to identify a wide range of LVCs. The *LF-INT* cases have a poor recall because the number of negative examples far outnumber the positive. The baseline accuracy for *LF-INT* also reflects this imbalance. We see a similar performance for individual light verbs in the news test set.

Individual LVs	LVs in test data	Precision	Recall	F1	Baseline	Accuracy
kar 'do'	650	96.54	95.07	95.80	81.23	93.23
ho 'be'	454	72.26	83.49	77.47	54.62	77.97
de 'give'	216	85.36	70.00	76.92	53.7	80.5
le 'take'	110	91.66	52.38	66.66	61.81	80
LF-TR	263	85.71	42.85	57.14	73.0	86.31
LF-INT	1064	83.33	16.12	27.02	88.34	89.84

Table 5: Precision, Recall and F1 for individual light verbs in the ICON test set, using Logistic Regression. The baseline accuracy uses the verb lemma as the feature.

5.1 Discussion

In order to understand the most informative features for our model, we examined the top 25 best performing features for the SVM model. We found that the best features for the positive class included the light verb lemma *kar* and a high-frequency noun lemma *shuru* 'begin'. Both log-likelihood and PMI were highly predictive of the positive class as well as the new feature using nominal argument postpositions. Semantic features such as 'Artifact,Object,Inanimate,Noun' were predictive of the negative class. This shows us that the linguistically motivated features are indeed effective for identification. From the analysis, it also appeared that semantic features overall can be more discriminative than lexical features for Hindi. This result is congruent with the results from English in Chen et al. (2015), who also use WordNet and sense annotated data as features.

We also made use of the probability scores for each class to understand the confidence of the classifier in assigning an instance to a positive or negative class. We found that in general, both classifiers were more confident in predicting the negative class label as their probabilities formed a distribution that was grouped closer to 1. The scores given to the positive class on the other hand were more distributed, with several instances that were less than 0.5. When we examined the LVCs with lower confidence scores, we saw that this was a mixed bag. For example, there were some LVCs like *bhojan kar* 'meal do; eat', which appeared to be non-LVCs. Others such as *photo le* 'photo take; take a photo' were perhaps cases of noun incorporation as suggested in Davison (2005). Still others were cases like *bojh daal* 'weight put; to be a burden (on someone)', which were more idiomatic in nature. This result shows that perhaps some of these cases are simply less frequent, but also that LVC annotation in the Hindi Treebank itself could be re-considered, or made more fine-grained based on the confidence scores of these models.

Our experiments show that LVCs in Hindi consist of diverse types that can be identified automatically using linguistic features. Unlike English, where the deverbal noun can be used as an important lexical indicator of 'lightness', in Hindi it becomes necessary to make use of other morpho-syntactic cues such as postpositions. However, it appears that the role of semantic features in general seems to be important for light verb identification in Hindi as well as English.

The models we have described in this paper show an improvement over previous work, but at the same time they can also be used to further refine the LVC annotation in the Hindi Treebank. This would give us more clarity with respect to the linguistic behaviour of these cases in Hindi and serve as a guideline

for LVC annotation in the future.

Acknowledgements

We gratefully acknowledge the support of the DST-CSRI (Department of Science and Technology (Govt. of India), Cognitive Science Research Initiative) fellowship funding titled 'Processing of complex event structures in Hindi: an investigation into relative compositionality' for the first author, as well as funding from CLEAR at the University of Colorado. This work was also supported by NSF grants CNS-0751202 and CNS-0709167. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Rafiya Begum, Karan Jindal, Ashish Jain, Samar Husain, and Dipti Misra Sharma. 2011. Identification of Conjunct Verbs in Hindi and their effect on Parsing Accuracy. In *In Proceedings of the 12th CICLing, Tokyo, Japan*.

Akshara Bharati, Dipti Misra Sharma, Samar Husain, Lakshmi Bai, Rafiya Begam, and Rajeev Sangal. 2012. AnnCorra : TreeBanks for Indian Languages. Technical Report Version-2.5, IIIT Hyderabad.

Pushpak Bhattacharyya, Debasri Chakrabarti, and Vaijayanthi Sarma. 2007. Complex Predicates in Indian languages and Wordnets. *Language Resources and Evaluation*, 40(3-4):331–355.

Pushpak Bhattacharyya. 2010. IndoWordNet. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, pages 3785–3792.

Miriam Butt, Tina Bögel, Annette Hautli, Sebastian Sulger, and Tafseer Ahmed. 2012. Identifying Urdu Complex Predication via Bigram Extraction. In *Proceedings of COLING 2012: Technical papers*, pages 409–424.

Miriam Butt. 2010. The Light Verb Jungle: Still Hacking Away. In M. Amberber, M. Harvey, and B. Baker, editors, *Complex Predicates in Cross-Linguistic Perspective*, pages 48–78. Cambridge University Press.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27.

Yi-Wei Chen and Chih-Jen Lin, 2006. *Combining SVMs with Various Feature Selection Strategies*, pages 315–324. Springer Berlin Heidelberg, Berlin, Heidelberg.

Wei-Te Chen, Claire Bonial, and Martha Palmer. 2015. English Light Verb Construction Identification Using Lexical Knowledge. In *Proceedings of the AAAI-15, Austin, TX, USA*.

Alice Davison. 2005. Phrasal predicates: How N combines with V in Hindi/Urdu. In Tanmoy Bhattacharya, editor, *Yearbook of South Asian Languages and Linguistics*, pages 83–116. Mouton de Gruyter.

R.-E. Fan, K.-W. Chang, C.-J. Hseih, X.-R. Wang, and C.-J. Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Spence Green, Marie-Catherine de Marneffe, and Christopher D Manning. 2013. Parsing Models for Identifying Multiword Expressions. *Computational Linguistics*.

Gregory Grefenstette and Simone Teufel. 1995. Corpus-based method for automatic identification of support verbs for nominalization. In *Proceedings of the 7th Meeting of the European Chapter of the Association for Computational Linguistics (EACL'95)*.

Samar Husain. 2009. Dependency Parsers for Indian languages. In *Proceedings of the ICON 2009 Tools Contest: Indian Language Dependency Parsing*.

Otto Jespersen. 1965. *A Modern English Grammar on Historical Principles, Part VI, Morphology*. George Allen and Unwin Ltd.

Yamuna Kachru. 2006. *Hindi*. John Benjamins.

Tara Mohanan. 1994. *Argument Structure in Hindi*. CSLI Publications, Stanford.

Dipak Narayan, Debasri Chakrabarti, Prabhakar Pande, and Pushpak Bhattacharyya. 2002. Experiences in Building the Indo WordNet- A WordNet for Hindi. In *First International Conference on Global WordNet, Mysore, India*.

Santanu Pal, Tanmoy Chakraborty, and Sivaji Bandopadhyay. 2011. Handling Multi-word expressions in Phrase-based Statisical Machine Translation. In *Proceedings of the Workshop on Multiword Expressions (MWE 2011), 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*.

Martha Palmer, Rajesh Bhatt, Bhuvana Narasimhan, Owen Rambow, Dipti Misra Sharma, and Fei Xia. 2009. Hindi Syntax: Annotating Dependency, Lexical Predicate-Argument Structure, and Phrase Structure. In *Proceedings of ICON-2009: 7th International Conference on Natural Language Processing*, Hyderabad.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Dhirendra Singh, Sudha Bhingardive, Kevin Patel, and Pushpak Bhattacharyya. 2015. Detection of Multiword Expressions for Hindi Language using Word Embeddings and WordNet-based Features. In *Proceedings of ICON-2015*.

Suzanne Stevenson, Afsaneh Fazly, and Ryan North. 2004. Statistical measures of the semi-productivity of light verb constructions. In *Proceedings of the 2nd ACL Workshop on Multiword Expressions: Integrating Processing*.

Sebastian Sulger and Ashwini Vaidya. 2014. Towards Identifying Hindi/Urdu Noun Templates in Support of a Large-Scale lfg Grammar. In *Proceedings of the Fifth Workshop on South and Southeast Asian Natural Language Processing at COLING 2014*.

Yee Fan Tan, Min-Yen Kan, and Hang Cui. 2006. Extending corpus based identification of light verb constructions using a supervised learning framework. In *Proceedings of the EACL 2006 Workshop on Multi-word-expressions in a multilingual context*.

Yuancheng Tu and Dan Roth. 2011. Learning English Light Verb Constructions: Contextual or Statistical. In *Proceedings of the Workshop on Multiword Expressions (MWE 2011), 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*.

Veronika Vincze, István Nagy T, and Gabór Berend. 2011. Detecting noun compounds and light verb constructions: a contrastive study. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011)*.