Building Tempo-HindiWordNet : A Resource for Effective Temporal Information Access in Hindi

Dipawesh Pawar[1], Mohammed Hasanuzzaman[2], Asif Ekbal[1]

[1]Department of Computer Science and Engineering
Indian Institute of Technology, Patna
{dipawesh.mtcs14,asif}@iitp.ac.in
[2]Normandie University, GREYC UMR 6072, Caen, France
Mohammed.Hasanuzzaman@unicaen.fr

Abstract

In this paper, we put forward a strategy that supplements Hindi WordNet entries with information on the temporality of its word senses. Each synset of Hindi WordNet is automatically annotated to one of the five dimensions: past, present, future, neutral and atemporal. We use semi-supervised learning strategy to build temporal classifiers over the glosses of manually selected initial seed synsets. The classification process is iterated based on the repetitive confidence based expansion strategy of the initial seed list until cross-validation accuracy drops. The resource is unique in its nature as, to the best of our knowledge, still no such resource is available for Hindi.

Keywords: Time sense annotation, Tempo-HindiWordNet, Semi-supervised learning

1. Introduction

There is considerable academic and commercial interest in processing time information in text, where the information is expressed either explicitly, or implicitly, or connotative. Recognizing such information and exploiting it for Information Retrieval (IR) and Natural Language Processing (NLP) tasks are important features that can significantly improve the functionality of these applications.

Whereas most of the prior computational linguistics and text mining temporal studies have focused on temporal expressions and events, there has been a lack of work looking at the temporal orientation of word senses. Towards this direction, (Dias et al., 2014) developed TempoWordNet (TWn), an extension of WordNet (Miller, 1995), where each synset is augmented with its temporal connotation (past, present, future or atemporal). It mainly relies on the quantitative analysis of the glosses associated to synsets, and on the use of the resulting vectorial term representations for semi-supervised synset classification. The authors show that improvements can be reached for temporal sentence classification and temporal query intent classification when unigrams are temporally expanded using the constructed TWn.

Being motivated from the idea of Dias et al. (Dias et al., 2014), we propose to build a temporal lexical knowledge-base (Tempo-HindiWordNet) which may contribute to the information access research in Hindi, a resource-scared language. Hindi is the national language of India and 4.46 % of world population is the native speaker of it. Semi-supervised learning strategy is used to automatically annotate each synset of Hindi WordNet (Bhattacharyya et al., 2010) to one of the five time-sensed classes: past, present, future, neutral and atemporal. One might suggests some synset do not clearly fall into one of the four classes (past, present, future, atemporal) proposed in (Dias et al., 2014). For example, does synset *sunday.n.01—first day of the week* have a predominant temporal class? Therefore, in addition to the four classes proposed by (Dias et al., 2014), we introduce another class as 'neutral'. Such synset would be marked as 'neutral' in our annotation scheme. Thus, one might consider our 'neutral' class to encompass past, present and future class. Classes, their meanings and examples are shown in Table 1. In the table, we use the ITRANS notation [1] for representing Hindi words.

The overall process is composed of two steps. In the first step, entire Hindi WordNet is classified in two classes, viz. temporal and atemporal. In second step the instances which are predicted as temporal in the preceding step are classified into four classes, viz. past, present, future and neutral.

Class	Meaning	Example
Past	already happened	कल(kala)
Present	currently going on	आज(AAja)
Future	about to happen	भावी(bhaavee)
Neutral	Overlapping time sense	संध्या(saNdhyaa)
Atemporal	absence of time sense	उष्ण(UShNNa)

Table 1: Example classes in HindiWordNet

The rest of the paper is organized as follows. In next section, we describe the related research on temporal information access. Problem statement and challenges are explained in Section 3., whereas the structure of Tempo-HindiWordNet is presented in section 4. We describe the methodology that we developed in Section 5. Results and discussions of the experiments for 10-fold cross-validation and on the gold standard test set are presented in Section 6. Finally, we conclude in Section 7.

---

[1]www.aczoom.com/itrans/

2.   Related Works

We begin by discussing two lines of research that are closely relevant to this work of associating temporality to word senses: 1) temporality in NLP and IR, and 2) linking words/senses with various cognitive features.

Temporality in NLP and IR: Temporality has recently received increased attention in Natural Language Processing (NLP) and Information Retrieval (IR). A good review of the initial works is exhaustively summarized in (Mani et al., 2005). The introduction of the TempEval task (Verhagen et al., 2009) and subsequent challenges (TempEval-2 and -3) in the Semantic Evaluation workshop series have clearly established the importance of time to deal with different NLP tasks. In IR, the time dimension has also received particular attention for the past few years. According to (Metzger, 2007), time is one of the key five aspects that determine a document's credibility besides relevance, accuracy, objectivity and coverage. So, the value of information or its quality is intrinsically time-dependent. As a consequence, a new research field called Temporal Information Retrieval (T-IR) has emerged and deals with all classical IR tasks such as crawling (Kulkarni et al., 2011), indexing (Anand et al., 2012) or ranking (Kanhabua et al., 2011) from the viewpoint of time. From an application perspective of T-IR, (Campos et al., 2014) proposed a solution for temporal classification of queries by identifying top relevant dates in web snippets with respect to a given implicit temporal query, and temporal disambiguation is performed through a distributional metric called GTE. Evaluation challenges such as NTCIR-11 Temporalia task (Joho et al., 2014) further pushed this idea and proposed to distinguish whether a given query is related to *past, recency, future* or *atemporal*.

Linking cognitive features to word senses: Several attempts have been made in both computational linguistics and cognitive science that build resources linking words with several cognitive features such as abstractness-correctness (Coltheart, 1981), sentiment (Esuli and Sebastiani, 2005a), imageability (Coltheart, 1981), and colors (Özbal et al., 2011). There are two prior approaches that attempt to understand the underlying temporal orientation of word senses. In (Hasanuzzaman et al., 2014; Dias et al., 2014), authors developed TempoWordNet (TWn), an extension of WordNet, where each synset is augmented with its temporal connotation (past, present, future, or atemporal). It mainly relies on the quantitative analysis of the glosses associated to synsets, and on the use of the resulting vectorial term representations for semi-supervised synset classification.

To the best of our knowledge, this is the first attempt to associate temporality to word senses in Hindi.

3.   Task Definition and Challenges

In our work we build a resource that can be effectively used for temporal information access. Given the Hindi WordNet, the task is to identify temporal information associated with each of the synsets present in it. We use four temporal senses (i.e past, presen, future, and neutral) and one atemporal (denoting non-temporal instances).

Polysemy phenomenon is one of the challenges for processing temporal annotation. Polysemous words can have different temporal annotation for each sense. One such word in Hindi is "कल (kala) " that may have the following temporal annotation:

- **Word :** कल (kala)

  Synset : अतीत (Ateeta), अतीतकाल (Ateetakaala),भूतकाल (bhootakaala), अतीत काल(Ateet kaala),कल (kala), भूत काल (bhoot kaala),गत काल (gat kaal),पिछला ज़माना (pichhalaa Zamaanaa),पूर्वकाल (poorvakaala)

  Gloss : बीता हुआ समय या काल (beetaa huAA samaya yaa kaala)

  Example sentence : "यह उपन्यास अतीत की घटनाओं पर आधारित है । (yah Upanyaas Ateet kee ghaTanaaON para AAdhaarit hai ;;)/कल की बातों को याद करके दुखी होना अच्छा नहीं । (kal kee baatoN ko yaad karake dukhee honaa Achchhaa naheeN ;;)"

  Temporal Annotation : Past

- **Word :**कल (kala)

  Synset :कल (kala)

  Gloss : आज के बाद आनेवाले पहले दिन को (AAj ke baad AAnevaale pahale din ko)

  Example sentence : "मैं कल घर जाऊँगा । (maiN kal ghara jaaOONngaa ;;)"

  Temporal Annotation : Future

---

4.   Structure of Tempo-HindWordNet

Each synset of our Tempo-HindiWordNet contains the following information: Offset, Word, PoS, Sense number, Gloss, Class and Prediction value. Offset is an unique identifier associated with each synset, PoS denotes the part-of- speech category of the target word, sense number denotes the number associated with the target synset in Hindi WordNet, gloss denotes the meaning associated with the synset, class denotes the predicted time sense of the synset by the temporal classifier and prediction value denotes the confidence with which annotated time sense is predicted. Finally, in the temporal resource that we build, out of total 143,874 synsets in Hindi WordNet[2], 127,843 synsets are tagged as atemporal whereas 3,037 synsets are tagged as past, 3,580 as present, 3,778 as future, and 5,628 as neutral. We show the structure of the resource with few examples in the following frame:

---

[2]According to Hindi WordNet API v1.2

Synset_Offset, Word, PoS, Sense number, Gloss, Class, Prediction_Value

- 8741, 11, ADJECTIVE, 1, दस से एक अधिक (das se Ek Adhik), atemporal, 0.99

- 27884, स्वार्थी(svaarthee), NOUN, 1, वह जो स्वार्थ से भरा हुआ हो या अपना मतलब निकालनेवाला हो (vah jo svaarth se bharaa huAA ho yaa Apanaa matalab nikaalanevaalaa ho), present, 0.969

- 4216, स्वप्न (svapn), NOUN, 1, सोते समय दिखाई देने वाला मानसिक दृश्य या घटना (sote samaya dikhaaEE dene vaalaa maanasik dRishya yaa ghaTanaa), neutral, 0.942

5. Methodology

We build Tempo-HindiWordNet using a semi-supervised learning approach where each synset of Hindi WordNet is classified to one of the five classes as mentioned earlier. We develop two strategies:

1. In our first strategy we build a multi-class classifier that directly classifies WordNet synsets into four temporal and one atemporal categories in single step. We call this as one-step classification framework.

2. The second strategy works in two-steps. In the first step we build a binary classifier that classifies WordNet synsets into temporal vs. atemporal. In the second step, we build a multi-class classifier to further classify the temporal synsets into four categories, namely past, present, future and neutral. This approach is called as two-step classification framework.

In both of these strategies, due to unavailability of labelled corpora, we explore semi-supervised learning strategy that learns iteratively starting with a very small set of seed entities. We create a list of seed entities that represent aforementioned classes. These seed words are represented by their respective gloss in the training set. We use different n-gram models for representing the glosses. These n-grams are weighted with their term frequencies. Initial set of seed words is then iteratively expanded using a confidence-based expansion strategy. Here, initial seed list is augmented by adding the instances which are predicted with high confidence by the classifier. Hence, we believe that our incremental learning process is always supported with good quality instances for training in each iteration. Once the expansion process stops, classifier built is used to annotate each synset of Hindi WordNet with time sense. We sketch the algorithmic steps in Algorithm 1. It composes of two key steps, viz. creation of initial seed list and confidence based expansion strategy.

Algorithm 1 Basic steps of the algorithm

1: Select initial set of seed words.
2: Initialize the current stratified cross-validation accuracy to 1.
3: Initialize previous stratified cross-validation accuracy to 0.
4: **while** *Current stratified cross-validation accuracy* ≥ *Previous stratified cross-validation accuracy* **do**
5:    **for** $i = 1 \rightarrow n$ **do**
      {n} denotes number of folds in which test set is divided.
6:       Create model over glosses of current set of seed words.
7:       **if** $i = 1$ **then**
8:          Update previous stratified cross-validation accuracy.
9:       **end if**
10:      **if** $i = n$ **then**
11:         Update Current stratified cross-validation accuracy.
12:      **else**
13:         Evaluate the model on test set i
14:         Expand training set using confidence based expansion strategy.
15:      **end if**
16:    **end for**
17: **end while**

5.1. Creation of initial seed list

Just as words with positive or negative connotation or denotation are used in sentiWordNet (Esuli and Sebastiani, 2005b), here we start with seed words representing the classes: past, present, future, neutral and atemporal. We show some of the examples in Table 2. Selection of such seed words is very important as semi-supervised learning greatly depends on its quality. Distribution of seed words among the various temporal classes is important in order to ensure that temporal classifier is not biased to any particular class. One of the ways to select such initial temporal seed words is to select the words which appear in subtree of word "समय (samaya)" in Hindi WordNet. But, almost all the words extracted from this subtree are found to belong to the noun PoS categories. However, there are significant number of non-noun words which denote or connote temporal senses. In order to ensure that we do not miss these words and our process is not biased to some certain PoS categories, we follow a very rigorous process while creating the seed list manually, and thereby gurantee to cover all kinds of PoS categories. We create a list of 85 seed words, out of which 37 are atemporal and the rest are temporal. Temporal entities are equally distributed among all the four temporal classes, namely past, present, future and neutral. In one-step classification scenario we consider the list of seed words that contain all these classes. In two-step classification framework we consider the seed words annotated with only two classes (temporal vs atemporal) in the first step, and then in the second step we con-

sider the seed words annotated with the fine-grained classes (four classes, i.e. past, present, future and neutral).

past		present	
word(sense no)	PoS	word(sense no)	PoS
कल(kal)(1)	adv	मौजूदा(3)	adj
गुजरा(2)	adj	वर्तमान(vartamaan)(2)	n
विगत(vigat)(2)	adj	आज कल(AAj kal)(1)	adv
अतीत(Ateet)(1)	n	आज(AAj)(1)	adv
जन्म(janm) (1)	n	आज(AAj)(2)	n
Future		**Neutral**	
word(sense no)	PoS	word(sense no)	PoS
पश्चात्(pashchaat)(1)	adv	इद्दत(Iddat)(1)	n
आगामी(1)	adj	नित्यप्रति(1)	adv
कल(kal)(2)	adv	धीमा(dheemaa)(1)	adv
कल(kal)(2)	n	ठोहर(Thohara)(1)	n
अपेक्षित(ApekShit)(2)	adj	अरसा(Arasaa)(1)	n
Atemporal			
word(Sense no)		PoS	
सर्वाधिक(sarvaadhik)(1)		adj	
अनिवासी_भारतीय(Anivaasee_bhaarateeya)(1)		n	
ताजुब_होना(taajjuba_honaa)(1)		v	
राष्ट्रीकरण(raaShTreekaraNN)(1)		n	
अस्मिता(Asmitaa)(4)		n	

Table 2: Few examples of manually selected seed words

Temporal(Randomly selected)			
Word(sense no)	PoS	Word(sense no)	PoS
प्रदूषित (pradooShit)(1)	adj	गेंद_बाहर_फेंकना(geNda_baahara_pheNkanaa)(1)	v
विधिलेख(vidhilekh)(1)	n	बजे(baje)(1)	adv
फटकन(phaTakan)(1)	n	ईसापूर्व(EEsaapoorv)(1)	adv
शोषित(shoShit)(2)	adj	फुर्सत_से(phursata_se)(1)	adv
भरा_होना(bharaa_honaa)(1)	v	नाटक(naaTak)(4)	adj
Atemporal(Randomly Selected)			
Word(sense no)	PoS	Word(sense no)	PoS
भग्नपाद(bhagnapaad)(1)	n	ऊपर(OOpara)(10)	adv
मीठानीम(meeThaaneem)(1)	n	खुला(khulaa)(5)	adj
नियुक्त_करना(niyukta_karanaa)(3)	v	चुकंदर(chukaNdara)(2)	n
इंफ्रारेड(INphraareD)(1)	adj	जमना(jamanaa)(3)	v
यहाँ_वहाँ(yahaaNn_vahaaNn)(1)	adv	सालना(saalanaa)(4)	v

Table 3: Some examples of automatically extracted seed words during expansion process: First step of two-step classification

5.2. Confidence based expansion strategy

We use a semi-supervised learning technique to learn temporal connotation associated with the word, if any. We create a training set over the glosses of the initial seed entities. The gloss of the seed words are retrieved from the Hindi WordNet API [3]. We train a Support Vector Machine (SVM) (Joachims, 2002) on this training set. Test set is composed of all the synsets extrcated from the Hindi WordNet. This is divided into 10 folds. At first we classify the first fold into temporal and atemporal classes[4].The instances with highest

---

[3]http://www.cfilt.iitb.ac.in/wordnet/webhwn/API_ downloaderInfo.php

[4]We describe the process with respect to our two-step classification strategy

confidence of prediction are given higher priority while expanding the initial seed data so as to keep connotational properties of initial seed entities intact. In every iteration we add the instances in such a way that the ratio of instances of different classes are maintained at par with the initial class distribution. This is to ensure that the classifier is not biased to any specific class. We define a threshold, and based on this value we select the instances to be added to the initial seed list. This threshold depends on the size of current seed word set, and we fix it through a series of experiments. The seed list, thus expanded, is treated as the training set for SVM learning in the next iteration. The classifier trained on this expanded training set is then evaluated on the second fold. This process continues till all folds are traversed, and we call this entire process as a cycle. At the end of a cycle we re-train SVM classifier with the expanded training set, and perform 10-fold cross-validation. If we observe that cross-validation accuracy at the end of any cycle is not as good as the cross-validation accuracy at the beginning, then we do not iterate the process further. Otherwise, we continue for the next cycle and repeat the same set of steps.

Past(randomly selected)	
Word(sense no)	PoS
अनबोल(Anabol)(4)	adj
आद्याक्षर(AAdyaakShara)(2)	n
अंदर(ANdara)(2)	adv
ठोकर_लगना(Thokara_laganaa)(2)	v
समय_समायोजित_करना(samaya_samaayojita_karanaa)(2)	v
Present(randomly selected)	
Word(sense no)	PoS
बाल_अपराध(baala_Aparaadh)(2)	n
स्वार्जित(svaarjit)(2)	adj
हरसम्भव(harasambhav)(2)	adv
आवेदन_कर्ता(AAvedana_kartaa)(2)	n
अभिनिवेशित(Abhiniveshit)(2)	adj
Future(randomly selected)	
Word(sense no)	PoS
सिलसिला(silasilaa)(6)	n
दृष्टिपटलविकृति(dRiShTipaTalavikRiti)(2)	adj
देवी(devee)(8)	n
बरसी(barasee)(2)	n
आहित(AAhit)(2)	adj
Neutral(randomly selected)	
Word(sense no)	PoS
दीर्घतपा(deerghatapaa)(2)	adj
उत्खाता(Utkhaataa)(3)	adj
अनुष्ठापक(AnuShThaapak)(2)	n
पूर्ववत्(poorvavat)(2)	adv
अंजना(ANjanaa)(3)	n

Table 4: Examples of automatically extracted seed words during expansion process: Second step of two-step classification

6.   Results and Discussions

In this section we present the experimental setups and results of experiments with necessary discussions and analysis. For experiments we use Weka[5] implementation of LibSVM (Chang and Lin, 2011). We use the default parameter set of Weka to perform experiments. Each sysnset is represented with its gloss. Gloss of synset is encoded as a vector of word n-grams weighted with their term frequencies.

6.1.   Evaluation of one-step classification

In this section we report the experimental results of one-step classification framework where the target instance is classified into past, present, future, neutral and atemporal classes in single step. In Table 5, we report the 10-fold cross-validation results of this framework. We stop execution after the fourth cycle of experiments as we did not observe any performance improvement.

Unigram Representation	Cycle 1	Cycle 2	Cycle 3	Cycle 4
precision	0.611	0.879	0.968	0.993
recall	0.541	0.873	0.968	0.993
F-measure	0.566	0.867	0.968	0.993

Table 5: Results of 10-fold cross-validation for one-step classification: Unigram model

A close look at the expanded training set reveals that the method faces difficulties in predicting synsets having hidden (or, implicit) temporal connotation. This is mainly because temporal synsets are very few in WordNet, and hence during expansion process it rapidly moves towards a skewed distribution with training set being more biased towards the atemporal categories.

6.2.   Evaluation of two-step classification

As mentioned earlier, the first step deals with a binary classification framework, whereas the second step deals with a multi-class classification framework. In the first step we build a SVM based binary classifier that discriminates temporal synsets from the atemporal ones. The instances classified as temporal in the first step are then passed to the second step for further classification into more fine grained temporal classes. Thus, errors in the first step could be propagated to the second step of two-step classification. In the first step, we develop various models based on n-gram representation such as Unigram (U), Bigram (B), Trigram (T), UnigramToBigram (UtB), UnigramToTrigram (UtT), BigramToTrigram (BtT). We perform 10-fold cross validation, and show the results of these models in Figure 1. Form the figure, we see that bigram model is the most effective model in comparison to the other models. Results of this model in terms of precision, recall, F-measure for each cycle are shown in Table 6. After the fourth cycle accuracy drops, and hence process of iterative learning stops. At the end of fourth cycle we obtain an expanded training set that consists of
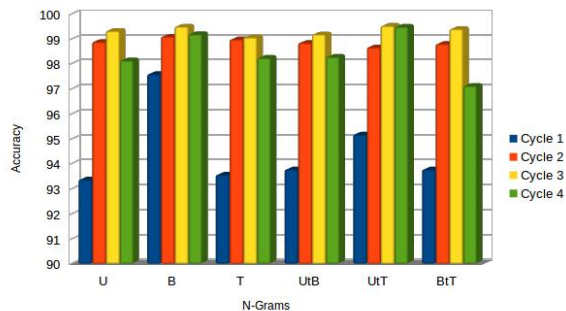
Figure 1: Results of various n-gram models: First step of two-step classification

34,325 instances. Out of these, 30,066 are atemporal and 4,259 are temporal. Some of the examples from the expanded set are shown in Table 3. It is seen that most of the temporal seed words are either adjectives (1,671) or nouns (2,449) and most of atemporal seed words are nouns (22,193). We observe that few non-temporal instances were wrongly classified to belong to the temporal categories. For example, as shown in Table 3 the word "भरा होना (bharaa honaa)" is wrongly classified as temporal. This might have attributed due to the ratio of temporal vs. atemporal instances, kept at the initial stage of the algorithm. In WordNet there are more atemporal instances as compared to temporals. Hence, it appears that we may be able to reduce such kinds of errors if we start with an initial configuration where number of atemporal instances would be even more (compared to the current setting) than the temporal instances.

Bigram Representation	Cycle 1	Cycle 2	Cycle 3	Cycle 4
precision	0.976	0.991	0.996	0.993
recall	0.976	0.991	0.996	0.993
F-measure	0.976	0.991	0.996	0.993

Table 6: Results of 10-fold cross validation for first step of two-step classification framework: Bigram model

We also use different n-gram models for finer classification, where a temporal instance is to be predicted with one of the four classes, namely past, present, future and neutral. Results of these various models are depicted in Figure 2.
Results show that SVM with bigram model achieves the highest accuracy, and therefore we select this model for further experiments.Its results in terms of precision, recall, F-measure for each cycle are reported in Table 7. The expanded training set, that we finally obtain, consists of 4,092 instances. Out of these there are 758, 752, 1,006, and 1,576 instances of past, present, future and neutral categories, respectively. Few example seed words added during expansion are shown in Table 4. From the expanded training set, it is observed that most of the seed words belonging to class neutral are either adjectives (1,671) or nouns
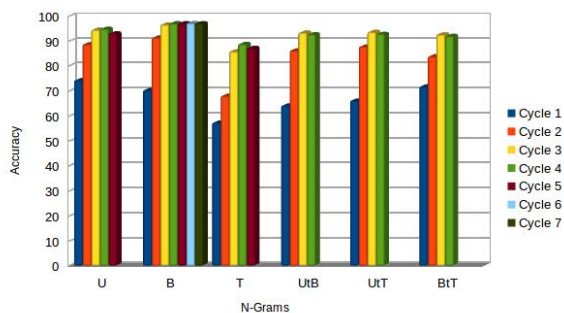
Figure 2: Results of different n-gram models: Second step of two-step classification

(2,449), for past mostly nouns (638), for present mostly adjectives (480) and for future mostly nouns (674).

Bigram Representation	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6	Cycle 7
Precision	0.715	0.957	0.989	0.99	0.99	0.99	0.99
Recall	0.703	0.955	0.989	0.989	0.99	0.99	0.99
F-measure	0.679	0.955	0.989	0.989	0.99	0.99	0.99

Table 7: Results of 10-fold cross validation for second step of two-step classification framework: Bigram model

6.3. Evaluation on gold standard test set

Due to the absence of temporally annotated corpus, we create a gold standard test set manually. It consists of 180 instances, 20 each for past, present, future, and neutral; and the rest 100 instances represent the atemporal class.

For one-step classification framework, we report 10-fold cross validation results in Table 8 where glosses are encoded as vector of word unigrams weighted with frequency counts.

Unigram Representation	precision	recall	F-measure
Past	0.267	0.2	0.228
Present	0.136	0.15	0.142
Future	0.142	0.05	0.073
Neutral	0.182	0.1	0.129
Atemporal	0.48	0.6	0.533
Avg	0.241	0.22	0.230

Table 8: Evaluation results of one-step Classification approach on gold standard test set

Thereafter, we evaluate the two-step classification framework on the gold standard test set. We use different n-gram variations, and results for first step are shown in Table 9. This shows that unigram model performs best for the temporal vs. atemporal classification over gold standard test set. Detailed evaluation results of the unigram representation are presented in Table 10.

The second step of the two-step algorithm is then evaluated on the gold standard test set. We train SVM

Representation	U	B	T	UtB	UtT	BtT
precision	**0.849**	0.598	0.833	0.829	0.642	0.818
recall	**0.82**	0.541	0.802	0.796	0.606	0.799
F-measure	**0.834**	0.568	0.817	0.812	0.623	0.809

Table 9: Evaluation results of various n-gram models for first step of two-step classification: Gold standard test set experiments

Unigram Representation	precision	recall	F-measure
Temporal	**0.903**	0.7	0.788
Atemporal	0.796	**0.94**	0.862
Avg.	0.849	0.82	0.834

Table 10: Results of the first step of two-step classification: Unigram model on gold standard test set

with various n-gram models, and its results are reported in Table 11. This shows that we achieve the best results for the bigram model. Based on this bigram model we produce more detailed evaluation for the individual classes, and its results are presented in Table 12.

Representation	U	B	T	UtB	UtT	BtT
precision	0.265	**0.659**	0.414	0.339	0.380	0.367
recall	0.275	**0.4**	0.337	0.287	0.325	0.337
F-measure	0.270	**0.497**	0.371	0.311	0.350	0.351

Table 11: Evaluation results of various n-gram models for the second step of two-step classification framework: gold standard test set experiments

It is evident from the results obtained through different experimental setups that temporal classifier, in general, performs remarkably well while we deal only with two classes, namely temporal and atemporal. However, results are not up to the mark while we attempt to perform classification with all the five classes (four temporal and one atemporal classes). Too stringent gold standard test set which hardly represents the training set might be one of the possible reasons for this. Training set does not cover the instances of all types of temporal classes. The use of more temporally-rich corpus such as newspaper text, narratives, etc. may be more useful as shown in the TempEval shared tasks (Verhagen et al., 2007).

6.3.1. Error Analysis

We perform error analysis both from the quantitative and qualitative perspectives.

One step classification: Confusion matrix showing the possible errors of the one-step classification framework is shown in Table 13. It shows that there are many false negatives, i.e. many temporal instances are

Bigram representation	precision	recall	F-measure
Past	**1.0**	0.25	0.4
Present	**1.0**	0.3	0.461
Future	0.334	0.05	0.086
Neutral	0.303	**1.0**	**0.465**
Avg	0.659	0.4	0.497

Table 12: Results of second step of two-step classification framework: Bigram model on gold standard test set experiments

wrongly predicted to belong to the atemporal class. This is because number of atemporal synsets in Word-Net are very high as compared to temporal synsets, and hence number of atemporal instances increases heavily in the training set as compared to temporal ones after each iteration during expansion. This causes training set to become more and more skewed towards atemporality. Thus, it makes it very difficult to keep properties of temporal seed words intact, which ultimately results in loss of quality of expansion process.

Classified as−>	Past	Present	Future	Neutral	Atemporal
Past	4	0	1	0	15
Present	1	3	0	0	16
Future	0	0	1	0	19
Neutral	1	0	2	2	15
Atemporal	9	19	3	9	60

Table 13: Confusion matrix of one step classification on gold standard data set

Classified as−>	Temporal	Atemporal
Temporal	56	24
Atemporal	6	94

Table 14: Confusion matrix of first step of two-step classification on gold standard data set : Unigram model

Two-step classification: We show the possible errors for the first step (i.e. temporal vs. atemporal) in confusion matrix as shown in Table 14. It shows that 30% of temporal synsets are wrongly predicted as atemporal. We observe that in these synsets temporal information are connotative. Temporal-atemporal classifiers built in this step are sometimes unable to capture such connotative temporality. Evaluation results presented in Table 9 show that unigram model performs better as compared to other n-gram models. This is because other higer order n-gram models check for extra information in the form of larger sequence matching. As an example, for the temporal instance such as word समय (samaya), the trigram model classifies the synset into temporal class only if words like

के पहले (ke pahale) ,के बाद (ke baad), etc. are present in the gloss of that synset along with the word समय (samaya). Hence, many instances where only temporal unigrams are present are not predicted to be temporal. Results reported in Table 10 show that precision is higher for the temporal class, i.e. there are less incorrect predictions. But, recall is higher for the atemporal class, which indicates that atemporal entities are correctly extracted. We observe that classifier was able to detect very unique entities such as समय (samaya), कल (kal), वाला (vaalaa), हुआ (huAA), etc. as the temporal ones. It, on the other hand, cannot detect some instances whose denotational meaning does not contribute to time sense, but connotational meaning does. For e.g. the word "ताजा (taajaa)" literally means something that is fresh but it's connotational meaning denotes present time sense. But such words were not detected to have temporal sense, and therefore recall drops compared to the atemporal instances.

Classified as−>	Past	Present	Future	Neutral
Past	5	0	2	13
Present	0	6	0	14
Future	0	0	1	19
Neutral	0	0	0	20

Table 15: Confusion matrix for second step of two-step classification on gold standard data set : Bigram model

Second step of two-step classification: We provide a quantitative analysis of the possible errors in Table 15. It shows that many synsets belonging to past, present or future are wrongly predicted to be neutral. This is an indication that our temporal classifier is probably biased towards neutral class. A closer analysis to the predicted instances shows that their finer temporal classes depend on emotional or cultural meaning associated with them and their glosses do not contain any entities which prominently provide any evidence for the finer temporal class. This, in turn, results in incorrect prediction. Looking at evaluation results for various n-gram models in Table 11, it is found that SVM with bigram representation performs better as compared to other n-gram models. One possible explanation behind this could be that multi-word instances play crucial role in predicting time senses. For e.g. unigrams like समय (samaya) might overlap over the classes past, present, future and neutral, but trigrams like समय के पहले (samaya ke pahale) ,समय के बाद (samaya ke baad) are not overlapping, i.e. they can not have more than one temporal sense. This may be one of the reasons behind the low accuracy with the unigram model that was not effective for classifying multi-word instances into past, present, future and neutral categories. Results of Table 12 show that precisions are higher for the past and present classes. This might be because instances of these classes such as "इस समय (Is samaya)", "चल रहा (chal rahaa)", "बीते हुए (beete huE)", etc. are non-overlapping with other

classes. Hence, these instances are properly detected. But, these are not the good candidates for generalization, and therefore, recall of these two classes are very low. For the neutral class, recall is higher; however, precision is lower as these instances often overlap with others, and therefore, results in incorrect classification.

7. Conclusion and Future work

In this paper we have presented a technique to automatically construct a temporal resource for Hindi language that could be effective for temporal information access. We have proposed different models based on semi-supervised learning framework that learns iteratively starting from a small set of seed entities. As a learning algorithm we use SVM, which was trained with different n-gram representations of glosses. We have presented an exhaustive evaluation framework where we report cross-validation accuracies as well as accuracies on a manually created gold standard test set.

We believe that our contribution towards building the temporal resource in Hindi will be an useful resource to the community, and will facilitate the research related to NLP and IR applications.

In future we would like to identify more features for the target task. In addition, we would like to explore different expansion strategies such as selection of instances on the basis of their semantic distance with respect to the instances currently present in the seed list. We would also like to investigate the use of word embedding and deep learning techniques for the task.

8. Bibliographical References

Anand, A., Bedathur, S., Berberich, K., and Schenkel, R. (2012). Index maintenance for time-travel text search. In *Proceedings of the 35th International ACM Conference on Research and Development in Information Retrieval (SIGIR)*, pages 235–244.

Campos, R., Dias, G., Jorge, A. M., and Jatowt, A. (2014). Survey of temporal information retrieval and related applications. *ACM Computing Survey*, 47(2):15:1–15:41.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Coltheart, M. (1981). The mrc psycholinguistic database. *The Quarterly Journal of Experimental Psychology*, 33(4):497–505.

Dias, G., Hasanuzzaman, M., Ferrari, S., and Mathet, Y. (2014). Tempowordnet for sentence time tagging. In *Companion Publication of the 23rd International Conference on World Wide Web Companion (WWW)*, pages 833–838.

Esuli, A. and Sebastiani, F. (2005a). Determining the semantic orientation of terms through gloss analysis. In *14th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 617–624.

Esuli, A. and Sebastiani, F. (2005b). Determining the semantic orientation of terms through gloss classification. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 617–624. ACM.

Hasanuzzaman, M., Dias, G., Ferrari, S., and Mathet, Y. (2014). Propagation strategies for building temporal ontologies. In *14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 6–11.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms.* Kluwer Academic Publisher.

Joho, H., Jatowt, A., Blanco, R., Naka, H., and Yamamoto, S. (2014). Overview of ntcir-11 temporal information access (temporalia) task. In *NTCIR-11 Conference (NTCIR)*, pages 429–437.

Kanhabua, N., Blanco, R., and Matthews, M. (2011). Ranking related news predictions. In *Proceedings of the 34th International ACM Conference on Research and Development in Information Retrieval (SIGIR)*, pages 755–764.

Kulkarni, A., Teevan, J., Svore, K., and Dumais, S. (2011). Understanding temporal query dynamics. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 167–176.

Mani, I., Pustejovsky, J., and Gaizauskas, R. (2005). *The language of time: a reader*, volume 126. Oxford University Press.

Metzger, M. (2007). Making sense of credibility on the web: Models for evaluating online information and recommendations for future research. *Journal of the American Society for Information Science and Technology*, 58(13):2078–2091.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Özbal, G., Strapparava, C., Mihalcea, R., and Pighin, D. (2011). A comparison of unsupervised methods to associate colors with words. In *Affective Computing and Intelligent Interaction*, pages 42–51. Springer.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). Semeval-2007 task 15: Tempeval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 75–80.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Moszkowicz, J., and Pustejovsky, J. (2009). The tempeval challenge: Identifying temporal relations in text. *Language Resources and Evaluation (LRE)*, 43(2):161–179.

9. Language Resource References

Pushpak Bhattacharyya and Prabhakar Pande and Laxmi Lupu. (2010). *Hindi Wordet. Language Resources and Evaluation (LRE)*, Lexicon, 1.0, ISLRN 853-261-507-123-4.