Cost and Benefit of Using WordNet Senses for Sentiment Analysis

Balamurali A R[†,‡], Aditya Joshi[‡], Pushpak Bhattacharyya[‡]

[†]IITB-Monash Research Academy, [‡]IIT Bombay
Mumbai, India-400076
balamurali@iitb.ac.in, aditya.jo@iitb.ac.in, pb@cse.iitb.ac.in

Abstract

Typically, *accuracy* is used to represent the performance of an NLP system. However, accuracy attainment is a function of investment in annotation. Typically, the more the amount and sophistication of annotation, higher is the accuracy. However, a moot question is *is the accuracy improvement commensurate with the cost incurred in annotation*? We present an economic model to assess the marginal benefit accruing from increase in cost of annotation. In particular, as a case in point we have chosen the Sentiment Analysis (SA) problem. In SA, documents normally are polarity classified by running them through classifiers trained on document vectors constructed from lexeme features, i.e., words. If, however, instead of words, one uses word *senses* (synset ids in wordnets) as features, the accuracy improves dramatically. But is this improvement significant enough to justify the cost of annotation? This question, to the best of our knowledge, has not been investigated with the seriousness it deserves. We perform a cost benefit study based on a *vendor-machine model*. By setting up a cost price, selling price and profit scenario, we show that although extra cost is incurred in sense annotation, the profit margin is high, justifying the cost. Additionally we show that a system that uses sense annotation achieves a break even point earlier than the system that does not use sense annotation. Our study is timely in the current NLP scenario of ML applied to richly annotated data, frequently obtained through crowd-sourcing which involves monetary investment.

Keywords: Cost Benefit, Sentiment Analysis, Annotation Cost

1. Introduction

A natural language processing (NLP) system consists of many sub-components. Often subcomponents that enhance the overall system performance require additional annotation. For example, it has been shown that WordNet senses are better features compared to lexeme based features for supervised sentiment classification task (Balamurali et al., 2011). In this case, annotating words with their WordNet senses is an additional task. Existing NLP research focuses on justifying alternate annotation approaches by showing their impact on performance of the system (often through improved accuracies). In this paper, we present a novel evaluation of improvement in an NLP system. Instead of the conventional way of evaluating by comparing their accuracies, we compare their *expected profits*. The novelty of this evaluation is two-fold:

1. Setting up expected profit of a system in terms of costs and expected returns.
2. Comparison of the approaches from different economic perspectives, namely which approach gives maximum expected profit and which approach gives this profit early.

Many important resources utilized in NLP applications require annotation. A silent objective of this work is to evaluate an annotation enabled system based on the effort spend to set it up for achieving the end performance. We find that even though NLP applications heavily depend on annotated resource, such a study, to the best of our knowledge, has never been done. The framework introduced will thus help in understanding whether deep linguistics and computational studies involving annotation should be carried out in practical scenarios in the light of escalating costs associated with it. With commercialization of annotation (in crowd-sourcing and other approaches), it is imperative to critically examine the value of annotation cost.

We use our cost-benefit model for evaluation of a sentiment analysis task which is enhanced by sense annotation to words.

1.1. Our Case Study

Sentiment classification deals with automatically tagging text as positive, negative or neutral with respect to a topic from the perspective of the speaker/writer. These categories are otherwise called as polarity classes. Thus, a sentiment classifier tags the sentence '*The museum is definitely worth a visit- serious fun for people of all ages!*' in a travel review as *positive*. On the other hand, a sentence '*The cafeteria is located in a dingy alley and one has to settle for stale food.*' is labeled as *negative*. Finally, '*The museum has twenty galleries on the first storey and sixteen on the second.*' is labeled as neutral. For the purpose of this work, we consider output labels as positive and negative according to the definition by Pang et al. (2002) & (Turney, 2002) and we show the variation in accuracy of sentiment classification using sense disambiguation for different values of training corpus size. Then, we proceed to the core of this work by presenting a cost-benefit model that justifies the cost of sense annotation using WordNet. In other words, we focus on answering the question:

> *Should the extra cost of sense annotation using WordNet be incurred for the sentiment classification task at all?*

To address the question, we describe a vendor-machine model that considers different machines, each of which uses one of the proposed approaches to sentiment classification. The model aims at helping a vendor arrive at a choice of which machine to buy (*i.e.* which approach to employ) in order to maximize her profits.

The rest of the paper is organized as follows. In Section 2., we first show that word senses are beneficial to sentiment classification. Our model for cost-benefit analysis is described in Section 3.. We justify additional investment in sense annotation using our model in Section 4.. Section 5. describes related work. Section 6. concludes the paper and points to future work.

2. Sense Disambiguation for Sentiment Classification

Our decision to use sentiment classification as a case study is motivated by Balamurali et al. (2011). The authors generate different variants of a travel review domain corpus by using automatic/manual sense disambiguation techniques. Thereafter, they compare classification accuracy of classifiers trained on different sense based and word based features. The experimental results show that WordNet senses act as better features as compared to words alone. For our analysis, we use classification engine by Balamurali et al. (2011) to generate results of these classifiers for different sizes of training corpus. We are not including details of experimental setup and dataset description due to lack of space. Please refer their paper for the same.

Depending on different approaches for sense annotation, we create the following corpora variants:

1. *Corpus with documents containing words.*
2. *Corpus with documents containing WordNet based synset identifiers corresponding to words. The words are annotated with senses by human annotators.*
3. *Corpus with documents containing WordNet based synset identifiers corresponding to words. In this case, the words are annotated with senses using a state-of-the-art WSD algorithm (Iterative WSD or in general IWSD) by Khapra et al. (2010a).*

Synset identifiers refer to WordNet offset[1] suffixed with POS category of the synset. Using each of these corpora, the following SVM[2] based classifiers are trained for the purpose of sentiment classification:

(a) *Classifier modeled with words as features (W)*

(b) *Classifier modeled with synset identifiers extracted from (2) as features (M)*

(c) *Classifier modeled with synset identifiers extracted from (3) as features (I)*

(d) *Classifier modeled with original words as well synset identifiers extracted from (2) as features (W+S (M))*

(e) *Classifier modeled with original words as well synset identifiers extracted from (3) as features (W+S (I))*

Classification accuracy obtained using different feature set is shown in Table 1[3]. The table also enumerates how the classifier accuracy varies as a function of training samples. Keeping in view that improved accuracy is a result of additional annotation (in this case, sense annotation using WordNet), the focus of our work is to validate the annotation cost vis-à-vis improvement in accuracy.

#Number of Training Documents	**W**	**M**	**I**	**W+S (M)**	**W+S (I)**
100	76.5	87	79.5	82.5	79.5
200	81.5	88.5	82	90	84
300	79.5	92	81	89.5	82
400	82	90.5	81	94	85.5
500	83.5	91	85	96	82.5

Table 1: Accuracy (%) with respect to number of training documents; W: Words, M: Manual Annotation, I: IWSD-based sense annotation, W+S(M): Word+Senses (Manual annotation), W+S(I): Word+Senses(IWSD-based sense annotation)

Assumptions: *5 sentences/review ; 12 words/sentence; 6 content words/sentence*

	Best case	**Average case**	**Worst case**
Number of sentences to be read in a review	Very few (1 sent)	Few (3 sent)	All (5 sent)
Sense annotation cost/review	0.3*6*5=9 units	0.3*6*5=9 units	0.3*6*5=9 units
Polarity annotation cost/review	0.3*6*1=1.8 units	0.3*6*3=5.4 units	0.3*6*5=9 units
Total cost/review	10.8 units	14.4 units	18 units

Table 2: Cost calculation for cases of input documents (for sense-based representation) based on the polarity annotation ease which in turn depends on number sentences to be analyzed. We fixed 6 as the number of content words based on the general observation of corpora we had.

3. Cost Model

We model cost associated with different approaches of sense annotation as costs to set up and to run five hypothetical machines. The costs impact the decision of a vendor to choose one of the options in Table 1 for providing a document level sentiment-annotation service. The model is detailed below in the form of a choice an NLP service provider has to make.

3.1. Vendor-machine scenario

A vendor wishes to provide a sentiment classification service. She knows of five off-the-shelf machines that perform sentiment classification. All the machines use a supervised technique. However, the machines differ in the feature representations used. The five machines use the following five feature representations: Word-based (W-machine), sense-based (manual) (M-machine), sense-based (automatic) (I-machine), words+senses (manual)(W+S(M)-machine) and words+senses (automatic) (W+S(I)-machine). These techniques were described in Section 2.. Note that except W-machine, all the other machines incur an additional cost

[1] Indexes to synonyms stored in wordnet based on their POS category

[2] http://www.csie.ntu.edu.tw/ cjlin/libsvm/

[3] Note that these values are for 100 blind test samples for each polarity class.

for sense annotation. This is because the documents that the vendor will receive from her clients are in the form of words, the vendor has to additionally annotate words with their senses in order to use a machine that accepts sense-based representation. The vendor wishes to decide *which of the five machines will give her the maximum benefit in terms of profit she makes by servicing polarity annotation*.

3.1.1. Cost assumption

There are two types of costs associated with each of the machines: a one-time *setup cost or fixed cost* and a *variable cost or running cost* (Samuelson, 1985). The setup cost is the cost of training the model. This includes manual *polarity annotation* and if required, *sense annotation* costs for creation of the training corpus. The running cost is the cost of obtaining polarity prediction for a document using the model trained. Depending on the nature of feature representation, the running cost may include an additional sense annotation cost.

Sense Annotation cost: We need to fix a annotation cost associated with marking the correct sense of a word. The sense annotation cost per word may be different in different parts of the world. For this paper, we fix the annotation cost as 0.3 units as the cost of sense annotating a word. This is as per the standard sense annotation cost prescribed by NLP association of India for their on-going IndoWordNet project (Bhattacharyya, 2010). The units mentioned is in Indian rupees[4].

Unlike sense annotation cost, we were not successful in obtaining a standard polarity annotation cost. Therefore, we model our polarity annotation cost based on different scenarios of input documents and their sense annotation cost. The details of the same are provided in the following subsection.

3.2. Input Scenarios

The clients submit documents to the vendor for sentiment prediction. The running cost depends on the nature of the submitted document. On the basis of their *ease of sentiment detection*, we classify documents into 3 categories:

1. **Best case** (*A well-structured document*): The document typically begins or ends with a summary that expresses the sentiment. The sentiment prediction of such a document would mean *sentiment prediction of these limited sentences*.
2. **Worst case** (*A free-form document*): The writer of the document describes multiple aspects. The sentiment prediction of such a document would mean an understanding of *overall sentiment derived from all sentences in the document.*
3. **Average case** (*A document mid-way between the best and the worst case*): A reader would need to go through *2-3 sentences* in the document to understand the sentiment expressed.

The polarity annotation cost is derived directly from the cost associated with sentences which are relevant. Thus, for a best case document, we consider the polarity annotation cost per document to be equivalent to the cost of annotating one sentence and derive this cost as (*per-unit cost of sense annotation*) * (*number of content words per sentence*) * (*number of sentiment-bearing sentences in the document*) = 0.3 * 6 * 1 = 1.8 units. The sense annotation cost crudely represents the cost of understanding the meaning of word in its context. We summarize each of these cases in Table 2. The costs shown are for systems that use semantic features because of which both polarity labeling cost and sense annotation cost contribute to the total setup cost.

3.3. Setting up Costs and Profit

To model profit, we construct a penalty/reward model to incentivize the vendor in case of correct prediction and penalize her in case of incorrect prediction. We devise the notion of a profit margin. A profit margin of x% for a vendor implies:

- For each correctly classified document, the vendor makes *a profit of x% of the setup cost* of the document. (*i.e.* the vendor charges *x*% of the cost incurred over and above the cost itself.)
- For each misclassified document, the vendor incurs *a loss of x% of the setup cost* of the document. (*i.e.* the vendor pays *x*% of the cost incurred to the client as redressal for misclassification.)

For the purpose of our modeling, we assume $1/3rd$ or (33%) as the profit margin[5]. However, we also perform our analysis for different values of profit margins and observe that our findings hold true for them as well.

For example, consider the case in which the setup cost for classifying a document given by the client is 30 units. Then, *the vendor charges 40 units from the client* if the document is correctly classified. However, if the document is incorrectly classified, *the vendor in turn pays to the client 10 units*. Note that in this case, the overall cost that the vendor incurs is the 30 units for polarity annotating the document, along with the 10 units that she pays to the client for misclassification.

We consider T training documents and P test documents (known as the *expected demand* in economics parlance) for each machine. A_i represents the accuracy of sentiment classification by Machine i. The accuracy values considered are as per our experimental results in Table 1. As an example case, we show the worst-case cost-benefit for M-machine. M-Machine uses senses as features but does not have any in-built sense annotation mechanism. The setup cost involves cost for polarity annotation and sense annotation cost. As an example, the profit calculation for worst case scenario for M-Machine is shown below:

Total price:

Total = Setup cost + Running cost

Setup cost
For T training documents,
Polarity annotation cost =9*T

[4] $1 = 45 Indian Rupees

[5] Based on information from http://en.allexperts.com/q/Manufacturing-1473/profit-margin.htm

Assumptions: T training documents, P test documents to be serviced, A_i accuracy for i-machine				
Machine	**Description**	**Setup cost**	**Running cost**	**Profit Equation**
W	Words as features	9T	0	$0.15A_W$P-3P-9T
M	Sense as features (manual)	9T+9T	9P	$0.3A_M$P-15P-18T
I	Sense as features(automatic)	9T+9T	0	$0.3A_I$P-6P-18T
W+S(M)	Words+senses (manual)	9T+9T	9P	$0.3A_{W+S(M)}$P-15P-18T
W+S(I)	Words+senses (automatic)	9T+9T	0	$0.3A_{W+S(I)}$P-6P-18T

Table 3: Profit calculation for different machines in the worst case scenario

					Best case scenario						
	Training on 100 documents						**Training on 500 documents**				
Test docs	**W**	**I**	**M**	**W+S (I)**	**W+S (M)**	**Test docs**	**W**	**I**	**M**	**W+S(I)**	**W+S (M)**
100	-730.5	-4329.0	-5094.0	-4329.0	-5175.0	**100**	10.5	90.0	-702.0	**45.0**	-612.0
200	-561.0	-3258.0	-4788.0	-3258.0	-4950.0	**200**	201.0	**1260.0**	-324.0	1170.0	-144.0
500	-52.5	-45.0	-3870.0	-45.0	-4275.0	**500**	772.5	**4770.0**	810.0	4545.0	1260.0
1000	795.0	**5310.0**	-2340.0	**5310.0**	-3150.0	**1000**	1725.0	**10620.0**	2700.0	10170.0	3600.0
5000	7575.0	**48150.0**	9900.0	**48150.0**	5850.0	**5000**	9345.0	**57420.0**	17820.0	55170.0	22320.0
10000	16050.0	**101700.0**	25200.0	**101700.0**	17100.0	**10000**	18870.0	**115920.0**	36720.0	111420.0	45720.0

Table 4: Profit for different machines in best case scenario; W: Words, I: IWSD, M: Manual Annotation, W+S(M): Word+Senses (Manual), W+S(I): Word+Senses(IWSD).

(Polarity annotation cost per document $*$ T)
Sense annotation cost =9*T
(Sense annotation cost per document $*$ T)
(Refer to Table 2)

Total setup cost = 9T + 9T = 18T

Running cost
For P test documents,
Sense annotation cost =9*P
(Sense annotation cost per document $*$ P)

Total = 18T+9P

Selling price:

$$\begin{aligned} \text{No of correctly classi-}\\ \text{fied documents} &= \frac{A}{100} * P \\ \text{Expected Reward} &= \frac{A}{100} * P * 24 \\ \text{No of misclassified doc-}\\ \text{uments} &= \frac{(100-A)}{100} * P \\ \text{Expected Penalty} &= \frac{100-A}{100} * P * 6 \end{aligned}$$

$$\begin{aligned} \text{Selling price} &= \frac{AP}{100} * 24 - \frac{(100-A)}{100} P * 6 \\ &= 0.3AP - 6P \end{aligned}$$

Expected Profit:

Selling Price $-$ *Total Price*

=0.3AP$-$15P$-$18T

Here for calculating expected reward for classifying one document correctly is 24 which includes 33% profit (6 units) over the setup cost(18 units).

Table 3 shows profit expression for processing worst case documents for different machines mentioned. The running cost is zero for W-machine, I-machine and W+S(I)-machine since no manual effort is required for a document when it is received for sentiment prediction. On the contrary, the documents for M-machine and W+S(M)-machine need to be manually annotated with their senses and hence the running cost of 9P is incurred.

4. Results and Discussions

4.1. Which machine makes the maximum profit?

Table 4, Table 5 and Table 6 give the results of profit generated for the three scenarios of input documents: Best case, Average case and Worst case respectively. We show the performance of the system for two different values of training corpus sizes: 100 and 500. We limit to these two values due to lack of space. The values in tables indicate the profit of a machine for a given training and test corpus size. These values are computed using the expressions in Table 3 (shown for worst case input documents only) by substituting the accuracy values given in Table 1. For example, in Table 4, the profit for W-machine trained on 100 documents and tested for 1000 documents is *795.0 units* in the best case scenario, *2385.0 units* in average case scenario and *7575.0 units* in worst case scenario. For each value of test document size, the figures in bold represent the best profit value obtained. Please note that worst case refers to documents for which polarity is difficult to asses and not expensive documents to procure.

In all three cases (Table 4, Table 5, Table 6), the machines which use sense representation using WordNet give higher profit over the W-machine (which uses word representation). For the training size 100, the W+S (I) and I machines[6] give maximum profit in all cases. For the training corpus size 500, I-machine gives maximum profit in all cases. This difference can be attributed to the different

[6] Both have same accuracy for training size 100.

			Average case scenario								
	Training on 100 documents						Training on 500 documents				
Test docs	**W**	**I**	**M**	**W+S (I)**	**W+S (M)**	**Test docs**	**W**	**I**	**M**	**W+S (I)**	**W+S (M)**
100	-2191.5	-5772.0	-6492.0	-5772.0	-6600.0	**100**	31.5	**120.0**	-636.0	60.0	-516.0
200	-1683.0	-4344.0	-5784.0	-4344.0	-6000.0	**200**	603.0	**1680.0**	168.0	1560.0	408.0
500	-157.5	-60.0	-3660.0	-60.0	-4200.0	**500**	2317.5	**6360.0**	2580.0	6060.0	3180.0
1000	2385.0	**7080.0**	-120.0	**7080.0**	-1200.0	**1000**	5175.0	**14160.0**	6600.0	13560.0	7800.0
5000	22725.0	**64200.0**	28200.0	**64200.0**	22800.0	**5000**	28035.0	**76560.0**	38760.0	73560.0	44760.0
10000	48150.0	**135600.0**	63600.0	**135600.0**	52800.0	**10000**	56610.0	**154560.0**	78960.0	148560.0	90960.0

Table 5: Profit for different machines in average case scenario; W: Words, I: IWSD, M: Manual Annotation, W+S(M): Word+Senses (Manual), W+S(I): Word+Senses(IWSD)

			Worst case scenario								
	Training on 100 documents						Training on 500 documents				
Test docs	**W**	**I**	**M**	**W+S (I)**	**W+S (M)**	**Test docs**	**W**	**I**	**M**	**W+S (I)**	**W+S (M)**
100	-52.5	-15.0	-690.0	-15.0	-825.0	**100**	-3547.5	-7050.0	-7770.0	-7125.0	-7620.0
200	795.0	**1770.0**	420.0	**1770.0**	150.0	**200**	-2595.0	-5100.0	-6540.0	-5250.0	-6240.0
500	3337.5	**7125.0**	3750.0	**7125.0**	3075.0	**500**	262.5	**750.0**	-2850.0	375.0	-2100.0
1000	7575.0	**16050.0**	9300.0	**16050.0**	7950.0	**1000**	5025.0	**10500.0**	3300.0	9750.0	4800.0
5000	41475.0	**87450.0**	53700.0	**87450.0**	46950.0	**5000**	43125.0	**88500.0**	52500.0	84750.0	60000.0
10000	83850.0	**176700.0**	109200.0	**176700.0**	95700.0	**10000**	90750.0	**186000.0**	114000.0	178500.0	129000.0

Table 6: Profit for different machines in worst case scenario; W: Words, I: IWSD, M: Manual Annotation, W+S (M): Word+Senses (Manual), W+S (I): Word+Senses (IWSD)

classification accuracy achieved by each of the machines for different sizes of the training corpus.

We observe that the classification accuracy does not play a role in early stages of service (*i.e.* for a lower number of test documents) but as the machine is used over a period of time, the rate of profit gained is accelerated based on the classification accuracy.

The initial lag in generating profit can be attributed to the larger setup cost of the system due to large training corpus size. For example, consider W+S(M) machine in the worst case scenario (Table 6). There is a difference of 14% in terms of classification accuracy for the machine trained on 100 documents as opposed to the machine trained on 500 documents. This difference manifests itself when 200 documents are tested on the machine. If the machine has been trained on 100 documents, a profit of *150.0 units* is achieved whereas the same system trained on 500 documents is running at a *loss* of *6240.0 units*.

As the number of documents serviced becomes 5000, the machine trained on 500 documents achieves a profit of *60000 units* while the machine trained on 100 documents achieves a profit of *46950 units*. The best value of profit for a machine trained with 500 documents is *129000 units* as opposed to a smaller profit of *95700 units* for the machine that gives the best profit after being trained on 100 documents.

As the expected demand (number of test documents) increases, the profit obtained for I-machine further rises over the profit achieved by the W-machine. This signifies that in case of a machine that will be used for a long period of time, it is the semantic representation that gives a higher benefit over the lexeme representation. The benefit, as we show, is in terms of the profit obtained over the additional cost incurred for semantic representation.

We hypothesize that the worst-case scenario pertains to a document containing *implicit sentiment* - where no single sentence is a direct clue to the sentiment of the document and where a deeper analysis of sentiment is required to understand the subtle opinion expressed. To detect such a sentiment, an overall understanding of the document is required. As per our definition of the worst-case scenario input documents, an individual has to read all sentences in the document to detect its polarity. Our results in Table 6 show that for a worst case scenario, I-machine and W+S (I)-machine provide a better performance and a higher profit as compared with W-machine. This reinforces the need for NLP resources that improve statistical learning methodologies.

4.2. Which machine begins to achieve profit the earliest?

The profit one can make is the driving force behind an enterprise. The previous section showed that I-machine can fetch better profit than W-machine. One parameter that is closely related with profit is the break-even point (BEP). BEP is a point at which there is neither profit nor loss. A desirable property of any business is to reach BEP as quickly as possible. In this section, we present BEP attained by different machines as a function of test documents to be serviced and the profit generated.

We define BEP for our model as the number of test documents that is required to attain a point of no profit and no loss. Different number of training documents and corresponding BEP for different machines are shown in Table 7. Values in boldface represent the best possible BEP for the given set of training documents. The numbers in parentheses indicate the accuracy obtained at the corresponding

Number of Training Documents	W	I	M	W+S (I)	W+S (M)
100	107 (76.5)	**101 (79.5)**	163 (87)	**101 (79.5)**	185 (82.5)
200	196 (81.5)	194 (82)	312 (88.5)	**188 (84)**	300 (90)
300	303 (79.5)	296 (81)	414 (93.5)	**291 (82)**	456 (89.5)
400	388 (82)	394 (81)	593 (90.5)	**367 (85.5)**	546 (94)
500	473 (83.5)	**462 (85)**	732 (91)	480 (82.5)	653 (96)

Table 7: Break-even points in terms of number of test documents for different machines for different number of training documents; Values in parentheses represent actual accuracies of a machine for a given training corpus size; W:Words, I: IWSD, M: Manual Annotation, W+S (M): Word+Senses (Manual), W+S (I): Word+Senses (IWSD)

BEP. For example, a W-machine trained on 100 documents performs with an accuracy of 76.5% and attains a BEP on servicing 107 test documents. BEP analysis gives an idea how fast each approach would attain a state of profit for the business.

The results show that the W+S (I) machine working with accuracy of 84% achieves BEP after processing 188 test documents when trained on 200 documents. This means that by buying the W+S (I) machine, a vendor can start making profits earlier as compared to using W-machine. For example, in case of 100 training documents, I-machine reaches BEP when approximately 101 documents are tested. W-machine attains a BEP only after processing 107 documents. M-machine requires 163 test documents to reach a BEP. Thus, either the I-machine or the W+S (I) machine achieves break-even point the earliest.

One may argue that the BEP is close for W-machine and W+S(I) machine (107 and 101 for W-machine and W+S (I)-machine respectively trained on 100 documents). However, there is a difference of 3% in their accuracies which further supports the benefit of W+S(I) machine over W-machine.

Also, consider the change in BEP and accuracy for a machine for increasing number of training corpus. As the training corpus size increases, the accuracy of the system increases. However, this also shifts the BEP of the machine further. This means that in case a high accuracy is expected, the vendor must expect a delay in BEP and this can be attributed to the setup cost.

4.3. What if the vendor had other profit margins on her mind?

The selection of 33% as the profit margin for our analysis was based on general observation[7]. To analyze the profits obtained by the machines at different profit margin values, we repeated our analysis for a set of *normal profit margins* (25%, 33%, 50%) and *abnormal profit margins* (66%, 75%, 100%) (Samuelson, 1985). As shown for profit margin 33%, we setup profit equations and generate profit values for different scenarios of input documents (worst/average/best case). We limit ourselves to a subset of results shown in Figure 1 due to lack of space. We assume a configuration of machines trained on 100 documents and used to service 500 worst case documents. For all values of profit margin that are considered, I-machine gives the maximum expected profit except in the case of 25% profit margin. For 25% profit margin, M-machine gives marginally higher profit than I-Machine. Note that in case of 0% profit margin (i.e. for a no-profit-no-loss business), M-machine requires more cost to run as compared to W-machine due to the additional cost incurred on sense annotation.

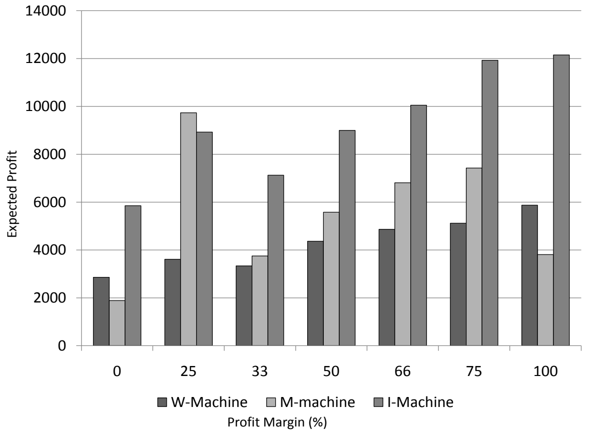

Figure 1: Expected profits(in INR) versus profit margin values for machines trained on 100 documents and used to service 500 worst case documents

4.4. A Note on NLP Parameters and Cost Benefit Analysis

We understand that to estimate the cost, we have not included many fine parameters required for building an NLP system such as cost for system runtime/compute power, cost for human experts setting up the system and supervising annotation *etc*. Their omission is not coincidental as the objective behind this paper was to introduce a novel yet a real world evaluation technique for NLP applications based on annotation. Our framework can be easily adapted to accommodate these parameters. We strongly believe that there is genuine need for a framework like ours because of an increasing cost associated with building state-of-the-art NLP applications and the gaining popularity of paid annotation using Crowd-sourcing.

5. Related Work

Though annotation provides an improvement in performance, the cost in obtaining the annotations is also a concern. One option to restrict the cost of annotation is to selectively annotate samples with high prediction uncertainty. This technique known as selective sampling (Blum

[7] Based on information from http://en.allexperts.com/q/Manufacturing-1473/profit-margin.htm

and Mitchell, 1998; Engelson and Dagan, 1996) aims at reducing the cost invested in annotation.

Our work focuses on the cost of annotation and presents an economic model to justify it through a cost-benefit analysis. Khapra et al. (2010b) perform an analysis similar to ours for a cross-lingual Word Sense Disambiguation (WSD) task to find an optimal balance between manual cross-linking and lexicon building in terms of value for accuracy. The authors model a metric based on reward/penalty to evaluate the improvement in performance and for cost incurred.

Our approach differs from the above work in two ways. The goal of cost-benefit analysis in the work is to find the optimal size of corpus, while we use it to compare different approaches (one without annotation, one with annotation) in order to justify the annotation. Hence, while the cost-benefit metric in their case uses a difference between gain over baseline and drop over upper bound, our cost-benefit metric is modeled as an expected profit based on accuracy of the underlying approach.

Dligach et al. (2010) also study the impact of annotation cost on the performance of a WSD application. The authors create two WSD systems- one using annotated data by one annotator and another by two annotators. A comparison of cost of annotation shows that a single annotator annotating more data for the same cost is likely to result in better system performance. The goal of their work is to study the effect of annotating more vis-à-vis annotating correctly to have a better system performance. The above work concentrates on proving the hypothesis that more data, although noisy, is probably better than having correct labeled data. In contrast to this work, we focus on establishing the worth of annotation in terms of economic benefit. In other words, while they address the question '*how much to annotate?*', we address a different question '*should we annotate? and how much gains will this annotation bring to me over a period of time?*'

While the related work has been in context of WSD, our work considers sentiment classification as a task and analyzes the cost-benefit of sense-annotation to it. This means that unlike the above-mentioned work, the annotation that we discuss is actually a component that is additional to the basic framework of the system. This makes it susceptible to analysis from the point of view of being an optional cost overhead. Our case study can be generalized to other applications where other performance-enhancing plugins that require annotation may have been added.

6. Conclusion and Future work

Our work studies the cost-benefit of NLP subcomponents or plugins that require additional annotation. In our case study, the performance-enhancing plugin is a WSD module added to a sentiment classification service. We represent different approaches as machines and show that the machines which use sense-based features using WordNet make higher profit than the one which uses word-based features. We calculate break-even points of all machines and show that the machines using sense-based features start reporting profit earlier. We also verify that our results hold true for different values of profit margin. Thus, we show that in spite of a cost overhead, there is an overall cost-benefit of sense annotation using WordNet to sentiment classification. We believe that our model forms a general framework to evaluate cost-benefit of additional annotation to any NLP task. However, there are limitations to our model. Many finer but necessary parameters of an NLP system are omitted in our study, it would be interesting to incorporate them and assess their contribution in achieving a break-even point. Further, our framework is not strong in modeling the cost parameters like how to fix the profit margin. While we fixed the profit margin based on general observations, it would be interesting to use optimization techniques to derive a profit margin which can reach a break-even point given a richer set of constraints.

7. Acknowledgments

We thank Jaya Saraswati and Rajita Shukla from CFILT Lab, IIT Bombay for annotating the dataset used for this work. We also thank Mitesh Khapra and Salil Joshi, IIT Bombay for providing us with the WSD engine.

8. References

AR Balamurali, Aditya Joshi, and Pushpak Bhattacharyya. 2011. Harnessing wordnet senses for supervised sentiment classification. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1081–1091, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Pushpak Bhattacharyya. 2010. Indowordnet. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

A Blum and T Mitchell. 1998. Combining labeled and unlabeled data with co-training. *Proceedings of COLT 98*, pages 92–100.

Dmitriy Dligach, Rodney D. Nielsen, and Martha Palmer. 2010. To annotate more accurately or to annotate more. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 64–72, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sean P. Engelson and Ido Dagan. 1996. Minimizing manual annotation cost in supervised training from corpora. In *Proceedings of ACL '96*, pages 319–326, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mitesh Khapra, Sapan Shah, Piyush Kedia, and Pushpak Bhattacharyya. 2010a. Domain-specific word sense disambiguation combining corpus based and wordnet based parameters. In *Proceedings of GWC-10*, Mumbai, India.

Mitesh M. Khapra, Saurabh Sohoney, Anup Kulkarni, and Pushpak Bhattacharyya. 2010b. Value for money: balancing annotation effort, lexicon building and accuracy for multilingual wsd. In *Proceedings of COLING '10*, pages 555–563. Association for Computational Linguistics.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. volume 10, pages 79–86. Association for Computational Linguistics.

Paul Anthony Samuelson. 1985. *Economics*. New York:McGraw-Hill, 12 edition.

Peter Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of ACL-02*, pages 417–424, Philadelphia, US.