Discriminative Reranking for Machine Translation

CS 712 Paper Presentation

Authors

Libin Shen, Anoop Sarkar, Franz Josef Och

Presented by

Amol Mandhane

Nivvedan S

Pankaj Singh

4 Apr 2014

Part I

Introduction

Introduction

Background

Generative models

- IBM models : finding $p(e|f)$ as combination of generative probability $p(e)$ and generative conditional probability $p(f|e)$
- Phrase based model: Phrase reordering and alignment

Discriminitive models

- Och and Ney (2002) proposed a framework for MT based on direct translation, using the conditional model $p(e|f)$ estimated using a maximum entropy model.
- A **small number of feature functions** defined on the source and target sentence were used to rerank the translations generated by a baseline MT system.

Introduction

Ranking and Reranking

- Ranking is a technique to remove scale information from different sets of data of same type. For example, translations for different sentences.
- Ranking process assigns **ranks** to the samples **based on some score**.
- Reranking is a technique to rank the samples again based on the output of some processing done on previously ranked samples.

Introduction

Discriminative reranking for MT

- The reranking approach for MT is defined as follows: First, a baseline system generates n-best candidates.
- Features that can **potentially discriminate between good vs. bad** translations are extracted from these n-best candidates.
- These features are then used to determine a new ranking for the n-best list. The new top ranked candidate in this n-best list is our new best candidate translation.

Introduction

Advantages of Discriminative Reranking

- Discriminative reranking allows us to use global features which are unavailable for the baseline system.
- Finally, the statistical machine learning approach has been shown to be effective in many NLP tasks.
- Reranking enables **rapid experimentation** with complex feature functions, because the complex decoding steps in SMT are done once to generate the N-best list of translations.

Introduction

Problems applying reranking to MT

- Unlike previous attempts in parse reranking, the algorithms cannot be directly applied to Machine Translation
- In MT, we have multiple reference translations as opposed to a single gold standard. Two candidate translations cannot be ranked accurately as one can be closer to ref_a and another to ref_b
- In terms of problem size, the size of the ranked list was 27 on an average for the parsing task. In MT, the size of the list is 1000
- Due to multiple reference translations, a strict ordering cannot be defined on the candidates

Introduction

Splitting and ordinal regression

Splitting

- We handle the problem by redefining the notion of good translations versus bad translations
- Instead of a strict ordering, we say that the **top r** of the n-best translations are good translations and the **bottom k** of them are bad translations
- We then look for **parallel hyperplanes** splitting the top r translations and the bottom k translations.

Ordinal regression

- However, by the above splitting, we lose the order information in the same class
- We might not care about distinguishing between e_{100} and e_{101}, but we might want to distinguish e_2 and e_{300} when $r = 300$
- Intuitively, this information seems useful and such insensitive ordering information is used

Introduction

Uneven margins and Large margin classifiers

Uneven margins

- Reranking is NOT an ordinal regression problem. In reranking, we're concerned only about the best translation and not the entire ordered set.
- For linear classifiers, we maintain a **larger margin in translations of high ranks** and smaller margin in translation of low ranks
- $margin(e_1, e_{30}) > margin(e_1, e_{10}) > margin(e_{21}, e_{30})$
- In keeping with the objective of reranking, penalty is decided.

Large margin classifiers

- SVMs are better suited for the task because of their ability of margin maximization
- However, SVMs are extremely slow in training for the task at hand.
- Therefore, a **perceptron-like** large-margin classification algorithm is used as they're very fast.
- Two such algorithms are proposed in this paper.

Part II

Reranking Algorithms

Splitting

The splitting algorithm finds a weight vector **w** perpendicular to a **separating hyperplane** which separates top r and bottom k translations for all sentences.

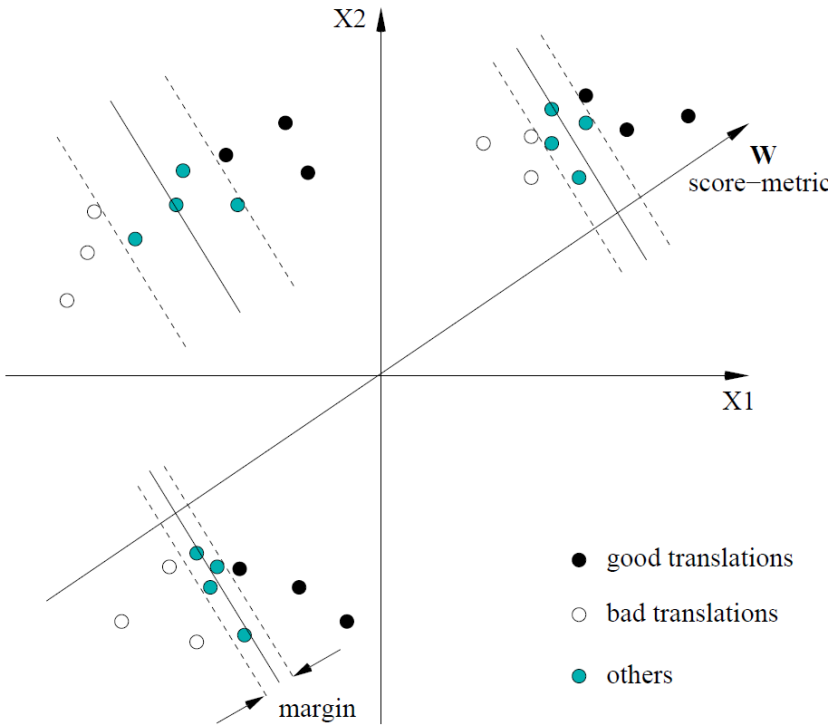

Figure 1 : Splitting for MT reranking

Splitting

Let $\mathbf{x}_{i,j}$ be the feature vector for the j^{th} translation for i^{th} sentence. Thus, the set of training examples becomes

$$S = \{(\mathbf{x}_{i,j}, y_{i,j}) \mid 1 \leq i \leq m,\ 1 \leq j \leq n\}$$

m is the number of sentences and n is the number of candidate translations for each sentence.

Construct a hypothesis function $h_f : \mathbb{R}^d \rightarrow \mathbb{R}$ with the separation function f (defined on next slide) to define the ranking as follows

$$(y_1^f, y_2^f, \ldots y_n^f) = h(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n) = rank(f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n))$$

Where rank is a function which takes a list of evaluation metrics and returns the ranks in that list.

$$rank(90, 40, 60) = (1, 3, 2)$$

Splitting

The splitting algorithm searched for a linear function $f(\mathbf{x}) = \mathbf{w}_f \cdot \mathbf{x}$ which successfully splits the top r-ranked and bottom k-ranked translations for each sentence, where $r + k \leq n$.

For top r-ranked sentences, $y_i \leq r$. For bottom k-ranked sentences, $y_i \geq n - k + 1$. We search for a function f which can separate good and bad translations after reranking. That is, the reranking output of f should be such that

$$y_i^f \leq r \quad if \quad y_i \leq r$$

$$y_i^f \geq n - k + 1 \quad if \quad y_i \geq n - k + 1$$

where y_i is the ranking done by evaluation metric.
The **minimal splitting margin**, γ^{split} for f is given by

$$\gamma^{split} = \min_i \left[\min_{y_{i,j} \leq r} f(\mathbf{x}_{i,j}) - \max_{y_{i,j} \geq n-k+1} f(\mathbf{x}_{i,j}) \right]$$

Splitting

Algorithm

Require: r, k and a positive learning margin τ

$t \leftarrow 0$, initialize $\mathbf{w}^0$

Repeat

for $(i = 1, \ldots, m)$ **do**

1. compute $\mathbf{w}^t \cdot \mathbf{x}_{i,j}$, $u_j \leftarrow 0$ for all j
2. **for** $(1 \leq j < l \leq n)$ **do**
   - 2.1 define $\{good, bad\} \equiv \{j, l\}$ such that $y_{i,good} \leq r$ and $y_{i,bad} \geq n - k + 1$, if they can be defined, else continue
   - 2.2 **if** $(\mathbf{w}^t \cdot (\mathbf{x}_{i,good} - \mathbf{x}_{i,bad}) < \tau)$ **then**
     - 2.2.1 $u_{good} \leftarrow u_{good} + 1$
     - 2.2.2 $u_{bad} \leftarrow u_{bad} - 1$
   - 2.3 **end if**
3. **end for**
4. $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \sum_j u_j \mathbf{x}_{i,j}$
5. $t \leftarrow t + 1$

end for

until no updates made in outer **for** loop

Splitting

Algorithm

The computational complexity of this algorithm is $\mathcal{O}(mn^2 + mnd)$. However, if the weight vector is updated every time an inconsistent pair is found, the complexity will be $\mathcal{O}(mn^2 d)$.

Theorem

Suppose the training samples $\{(\mathbf{x}_{i,j}, y_{i,j})\}$ *are splittable by a linear function defined on the weight vector* $\mathbf{w}^*$ *with a splitting margin* γ*, where* $\|\mathbf{w}^*\| = 1$*. Let* $R = \max_{i,j} \|\mathbf{x}_{i,j}\|$*. Then Algorithm 1 makes at most* $\frac{n^2 R^2 + 2\tau}{\gamma^2}$ *mistakes on the pairwise samples during the training.*

Above theorem implies that the splitting algorithm will **converge in finite iterations**.

Ordinal regression with uneven margins

Previously, we discussed the **importance of the ordering information** in reranking. Here, an ordinal regression algorithm is proposed which searches for w with **ordering information and uneven margins**.

The algorithm which is proposed in the ϵ-insensitive ordinal regression. In this algorithm, the function *dis* is used to control the **level of insensitivity**. The function g is used to control the learning margin between pairs of translations at different ranks for uneven margins. A simplest definition of g can be

$$g(p, q) = \frac{1}{p} - \frac{1}{q}$$

A simplest definition of *dis* can be

$$dis(p, q) = |p - q|$$

Ordinal regression with uneven margins

Algorithm

Require: a positive learning margin τ

$t \leftarrow 0$, initialize $\mathbf{w}^0$

Repeat

for $(i = 1, \ldots, m)$ **do**

1. compute $\mathbf{w}^t \cdot \mathbf{x}_{i,j}$, $u_j \leftarrow 0$ for all j
2. **for** $(1 \leq j < l \leq n)$ **do**
   - 2.1 define $\{less, more\} \equiv \{j, l\}$ such that $y_{i,less} < y_{i,more}$
   - 2.2 **if** $(dis(y_{i,less}, y_{i,more}) > \epsilon$ and $\mathbf{w}^t \cdot (\mathbf{x}_{i,less} - \mathbf{x}_{i,more}) < g(y_{i,less}, y_{i,more})\tau)$ **then**
     - 2.2.1 $u_{less} \leftarrow u_{less} + g(y_{i,less}, y_{i,more})$
     - 2.2.2 $u_{more} \leftarrow u_{more} - g(y_{i,less}, y_{i,more})$
   - 2.3 **end if**
3. **end for**
4. $\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t + \sum_j u_j \mathbf{x}_{i,j}$
5. $t \leftarrow t + 1$

end for

until no updates made in outer **for** loop

Part III

Experiments and Analysis

Experiments

Data

- Baseline MT Training data: about 170M English words
- Dev data: 993 Chinese sentences with 1000-best English translations
- Test data: 878 Chinese sentences with 1000-best English translations

Features

- Baseline – Sentence length, language models, lexical relationships, grammatical dependencies, phrase penalty, word penalty
- Best Feature – 6 Baseline + IBM Model 1 + matched parentheses + matched quotation marks + POS language model

More sets of features can be formed.

Results

Algorithm	Baseline	Best Feat.	Feat. Comb.
MERT	31.6	32.6	32.9
Splitting	31.7	32.8	32.6
Regression	31.4	32.7	32.9

Table 1 : Comparison between MERT and discriminative reranking on test data (BLEU %)

Results

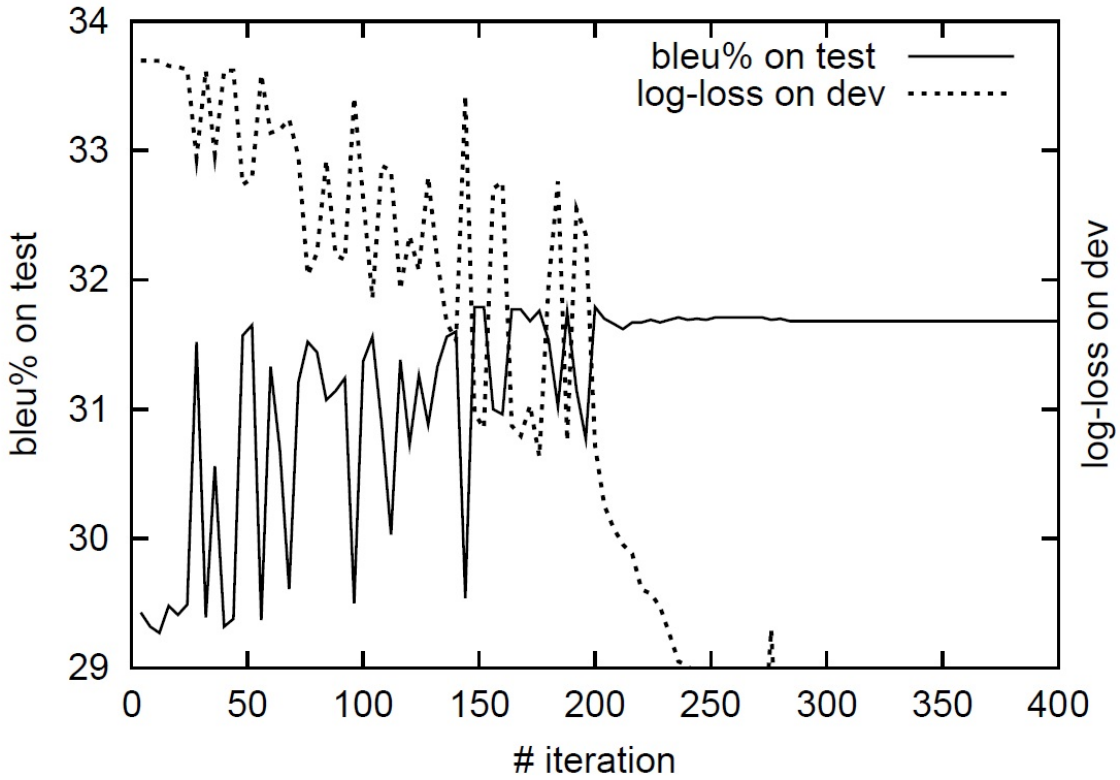

Figure 2 : Splitting on baseline

Results

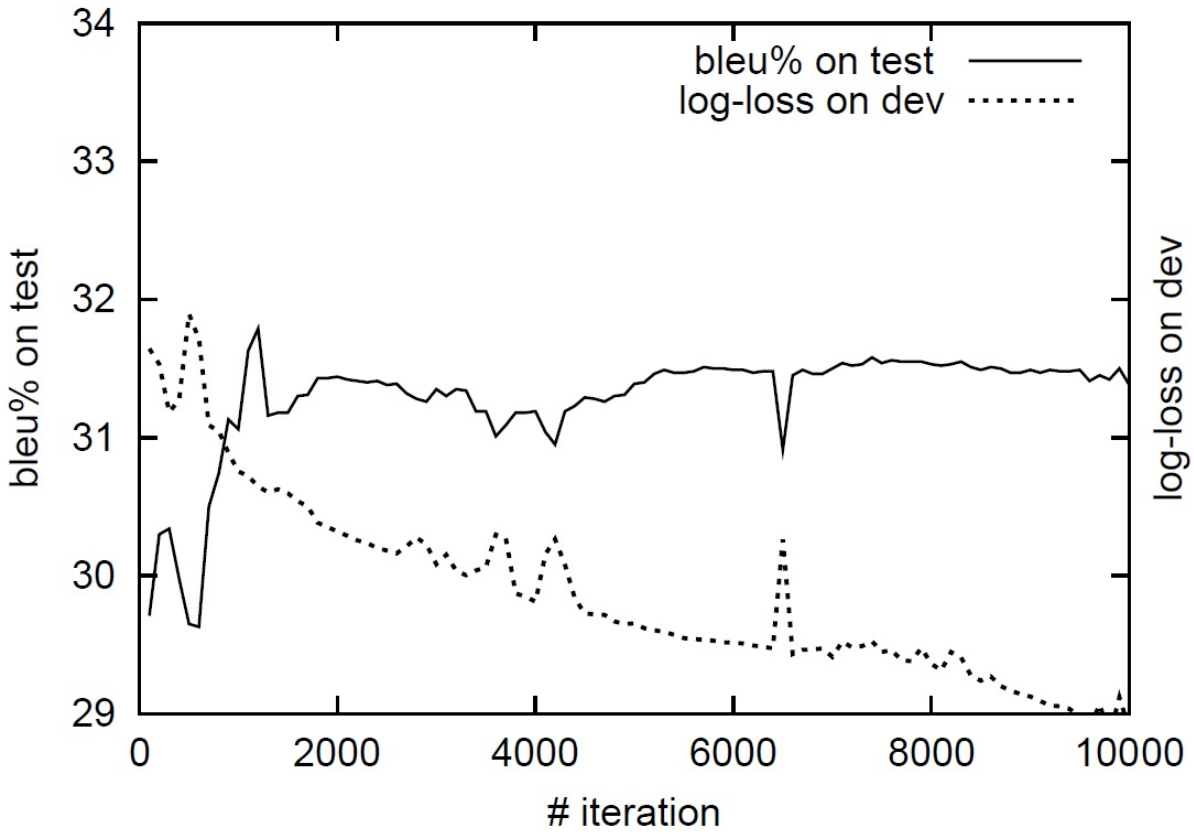

Figure 3 : Ordinal regression on baseline

Results

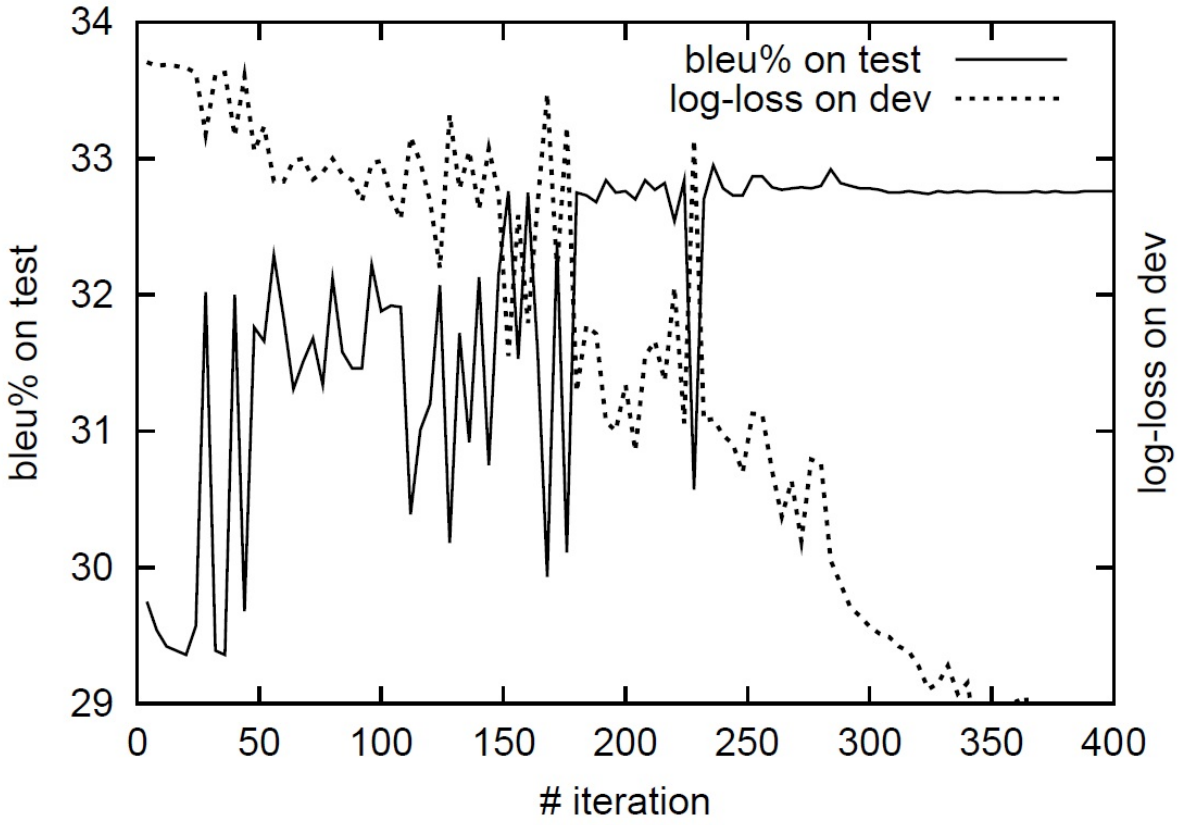

Figure 4 : Splitting on best feature set

Results

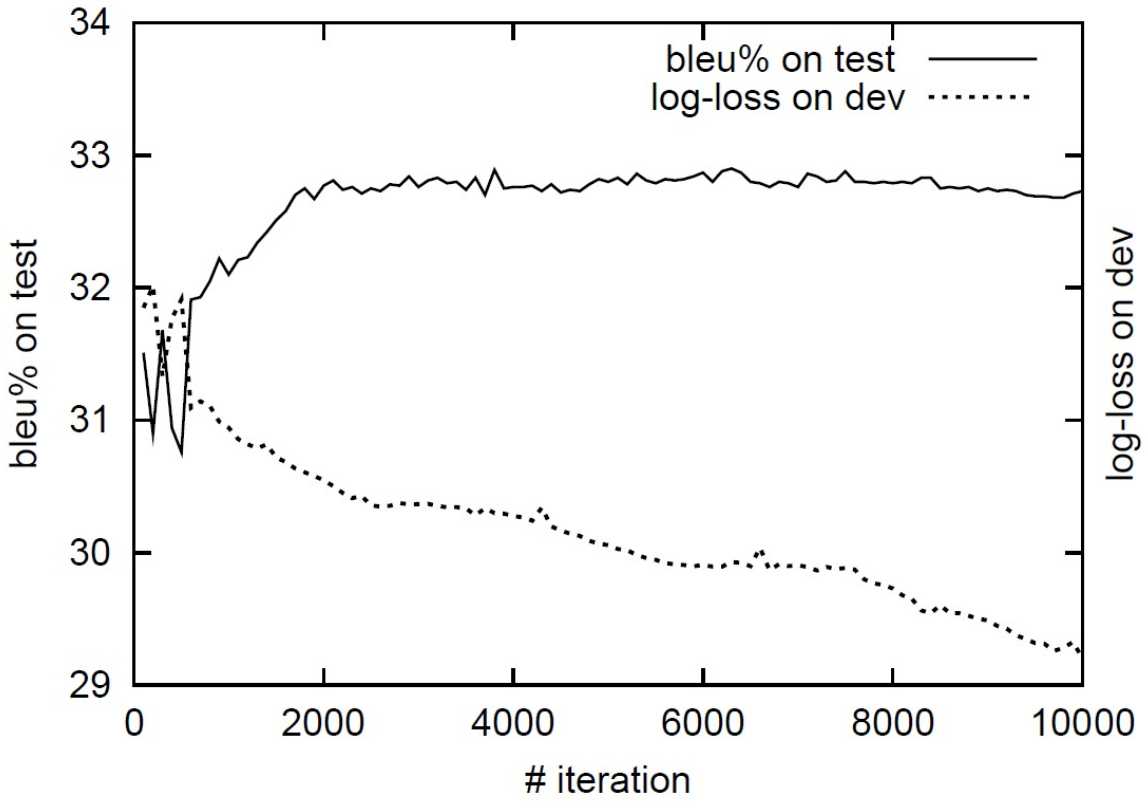

Figure 5 : Ordinal regression on best feature set

References

1. L. Shen, A. Sarkar, F. J. Och. 2004. Discriminative Reranking for Machine Translation. In *HLTNAACL 2004*.
2. M. Collins. 2000. Discriminative reranking for natural language parsing. In *Proceedings of the 7th ICML*.
3. R. Herbrich, T. Graepel, and K. Obermayer. 2000. Large margin rank boundaries for ordinal regression. In A.J. Smola, P. Bartlett, B. Scholkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 115132. MIT Press.
4. F. J. Och and H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *ACL 2002*.
5. F. J. Och. 2003. Minimum error rate training for statistical machine translation. In *ACL 2003*.

Thank you

Amol Mandhane
Nivvedan S
Pankaj Singh