Confidence-Weighted Learning of Factored Discriminative Language Models

Viet Ha-Thuc
Computer Science Department
The University of Iowa
Iowa City, IA 52241, USA
hviet@cs.uiowa.edu

Nicola Cancedda
Xerox Research Centre Europe
6, chemin de Maupertuis
38240 Meylan, France
Nicola.Cancedda@xrce.xerox.com

Abstract

Language models based on word surface forms only are unable to benefit from available linguistic knowledge, and tend to suffer from poor estimates for rare features. We propose an approach to overcome these two limitations. We use factored features that can flexibly capture linguistic regularities, and we adopt confidence-weighted learning, a form of discriminative online learning that can better take advantage of a heavy tail of rare features. Finally, we extend the confidence-weighted learning to deal with label noise in training data, a common case with discriminative language modeling.

1 Introduction

Language Models (LMs) are key components in most statistical machine translation systems, where they play a crucial role in promoting output fluency.

Standard n-gram generative language models have been extended in several ways. Generative factored language models (Bilmes and Kirchhoff, 2003) represent each token by multiple factors – such as part-of-speech, lemma and surface form– and capture linguistic patterns in the target language at the appropriate level of abstraction. Instead of estimating likelihood, discriminative language models (Roark et al., 2004; Roark et al., 2007; Li and Khudanpur, 2008) directly model fluency by casting the task as a binary classification or a ranking problem. The method we propose combines advantages of both directions mentioned above. We use factored features to capture linguistic patterns and discriminative learning for directly modeling fluency. We define highly overlapping and correlated factored features, and extend a robust learning algorithm to handle them and cope with a high rate of label noise.

For discriminatively learning language models, we use confidence-weighted learning (Dredze et al., 2008), an extension of the perceptron-based online learning used in previous work on discriminative language models. Furthermore, we extend confidence-weighted learning with soft margin to handle the case where training data labels are noisy, as is typically the case in discriminative language modeling.

The rest of this paper is organized as follows. In Section 2, we introduce factored features for discriminative language models. Section 3 presents confidence-weighted learning. Section 4 describes its extension for the case where training data are noisy. We present empirical results in Section 5 and differentiate our approach from previous ones in Section 6. Finally, Section 7 presents some concluding remarks.

2 Factored features

Factored features are n-gram features where each component in the n-gram can be characterized by different linguistic dimensions of words such as surface, lemma, part of speech (POS). Each of these dimensions is conventionally referred to as a *factor*.

An example of a factored feature is "pick PRON up", where PRON is the part of speech (POS) tag for pronouns. Appropriately weighted, this feature can capture the fact that in English that pattern is often fluent. Compared to traditional surface n-gram features like "pick her up", "pick me up" etc., the feature "pick PRON up" generalizes the pattern better. On the other hand, this feature is more precise

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:shortpapers, pages 439–444,
Portland, Oregon, June 19-24, 2011.

POS	Extended POS
Noun	SingNoun, PlurNoun
Pronoun	Sing3PPronoun, OtherPronoun
Verb	InfVerb, ProgrVerb, SimplePastVerb, PastPartVerb, Sing3PVerb, OtherVerb

Table 1: Extended tagset used for the third factor in the proposed discriminative language model.

than the corresponding POS n-gram feature "VERB PRON PREP" since the latter also promotes undesirable patterns such as "pick PRON off" and "go PRON in". So, constructing features with components from different abstraction levels allows better capturing linguistic patterns.

In this study, we use tri-gram factored features to learn a discriminative language model for English, where each token is characterized by three factors including surface, POS, and extended POS. In the last factor, some POS tags are further refined (Table 1). In other words, we will use all possible trigrams where each element is either a surface from, a POS, or an extended POS.

3 Confidence-weighted Learning

Online learning algorithms scale well to large datasets, and are thus well adapted to discriminative language modeling. On the other hand, the perceptron and *Passive Aggressive (PA)* algorithms[1] (Crammer et al., 2006) can be ill-suited for learning tasks where there is a long tail of rare significant features as in the case of language modeling.

Motivated by this, we adopt a simplified version of the CW algorithm of (Dredze et al., 2008). We introduce a score , based on the number of times a feature has been obseerved in training, indicating how confident the algorithm is in the current estimate w_i for the weight of feature i. Instead of equally changing all feature weights upon a mistake, the algorithm now changes more aggressively the weights it is less confident in.

At iteration t, if the algorithm miss-ranks the pair of positive and negative instances (p_t, n_t), it updates the weight vector by solving the optimization in Eq. (1):

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}} \frac{1}{2}(\mathbf{w}-\mathbf{w}_t)^\top \Lambda_t^2 (\mathbf{w}-\mathbf{w}_t) \quad (1)$$

$$\text{s.t.} \quad \mathbf{w}^\top \Delta_t \geq 1 \quad (2)$$

where $\Delta_t = \phi(p_t) - \phi(n_t)$, $\phi(x)$ is the vector representation of sentence x in factored feature space, and Λ_t is a diagonal matrix with confidence scores.

The algorithm thus updates weights aggressively enough to correctly rank the current pair of instances (i.e. satisfying the constraint), and preserves as much knowledge learned so far as possible (i.e. minimizing the weighted difference to $\mathbf{w}_t$). In the special case when $\Lambda_t = I$ this is the update of the Passive-Aggressive algorithm of (Crammer et al., 2006).

By introducing multiple confidence scores with the diagonal matrix Λ, we take into account the fact that feature weights that the algorithm has more confidence in (because it has learned these weights from more training instances) contribute more to the knowledge the algorithm has accumulated so far than feature weights it has less confidence in. A change in the former is more risky than a change with the same magnitude on the latter. So, to avoid over-fitting to the current instance pair (thus generalize better to the others), the difference between $\mathbf{w}$ and $\mathbf{w_t}$ is weighted by confidence matrix Λ in the objective function.

To solve the quadratic optimization problem in Eq. (1), we form the corresponding Lagrangian:

$$L(\mathbf{w},\tau) = \frac{1}{2}(\mathbf{w}-\mathbf{w}_t)^\top \Lambda_t^2 (\mathbf{w}-\mathbf{w}_t) + \tau(1-\mathbf{w}^\top \Delta) \quad (3)$$

where τ is the Lagrange multiplier corresponding to the constraint in Eq. (2). Setting the partial derivatives of L with respect to $\mathbf{w}$ to zero, and then setting the derivative of L with respect to τ to zero, we get:

$$\tau = \frac{1-\mathbf{w_t}^\top \Delta}{\|\Lambda^{-1}\Delta\|^2} \quad (4)$$

Given this, we obtain Algorithm 1 for confidence-weighted passive-aggressive learning (Figure 1). In the algorithm, P_i and N_i are sets of fluent and non-fluent sentences that can be contrasted, e.g. P_i is a set of fluent translations and N_i is a set of non-fluent translations of a same source sentence s_i.

[1] The popular MIRA algorithm is a particular PA algorithm, suitable for the linearly-separable case.

Algorithm 1 Confidence-weighted Passive-Aggressive algorithm for re-ranking.

Input: Tr $= \{(P_i, N_i), 1 \leq i \leq K\}$
$\mathbf{w}_0 \leftarrow 0, t \leftarrow 0$
for a predefined number of iterations **do**
  for i from 1 to K **do**
    for all $(p_j, n_j) \in (P_i \times N_i)$ **do**
      $\Delta_t \leftarrow \phi(p_j) - \phi(n_j)$
      if $\mathbf{w}_t^\top \Delta_t < 1$ **then**
        $\tau \leftarrow \frac{1 - \mathbf{w}_t^\top \Delta_t}{\Delta_t^\top \Lambda_t^{-2} \Delta_t}$
        $\mathbf{w}_{t+1} \leftarrow \mathbf{w_t} + \tau \Lambda_t^{-2} \Delta_t$
      Update Λ
      $t \leftarrow t + 1$
return $\mathbf{w}_t$

The confidence matrix Λ is updated following the intuition that the more often the algorithm has seen a feature, the more confident the weight estimation becomes. In our work, we set Λ_{ii} to the logarithm of the number of times the algorithm has seen feature i, but alternative choices are possible.

4 Extension to soft margin

In many practical situations, training data is noisy. This is particularly true for language modeling, where even human experts will argue about whether a given sentence is fluent or not. Moreover, effective language models must be trained on large datasets, so the option of requiring extensive human annotation is impractical. Instead, collecting fluency judgments is often done by a less expensive and thus even less reliable manner. One way is to rank translations in n-best lists by NIST or BLEU scores, then take the top ones as fluent instances and bottom ones as non-fluent instances. Nonetheless, neither NIST nor BLEU are designed directly for measuring fluency. For example, a translation could have low NIST and BLEU scores just because it does not convey the same information as the reference, despite being perfectly fluent. Therefore, in our setting it is crucial to be robust to noise in the training labels.

The update rule derived in the previous section always forces the new weights to satisfy the constraint (*Corrective* updates): mislabeled training instances could make feature weights change erratically. To increase robustness to noise, we propose a soft margin variant of confidence-weighted learning. The optimization problem becomes:

$$\arg\min_{\mathbf{w}} \frac{1}{2}(\mathbf{w} - \mathbf{w}_t)^\top \Lambda_t^2 (\mathbf{w} - \mathbf{w}_t) + C\xi^2 \quad (5)$$

$$\text{s.t.} \qquad \mathbf{w}^\top \Delta_t \geq 1 - \xi \quad (6)$$

where C is a regularization parameter, controlling the relative importance between the two terms in the objective function. Solving the optimization problem, we obtain, for the Lagrange multiplier:

$$\tau = \frac{1 - \mathbf{w_t}^\top \Delta_t}{\Delta_t^\top \Lambda_t^{-2} \Delta_t + \frac{1}{2C}} \quad (7)$$

Thus, the training algorithm with soft-margins is the same as Algorithm 1, but using Eq. 7 to update τ instead.

5 Experiments

We empirically validated our approach in two ways. We first measured the effectiveness of the algorithms in deciding, given a pair of candidate translations for a same source sentence, whether the first candidate is more fluent than the second. In a second experiment we used the score provided by the trained DLM as an additional feature in an n-best list reranking task and compared algorithms in terms of impact on NIST and BLEU.

5.1 Dataset

The dataset we use in our study is the Spanish-English one from the shared task of the WMT-2007 workshop[2].

Matrax, a phrase-based statistical machine translation system (Simard et al., 2005), including a trigram generative language model with Kneser-Ney smoothing. We then obtain training data for the discriminative language model as follows. We take a random subset of the parallel training set containing 50,000 sentence pairs. We use Matrax to generate an n-best list for each source sentence. We define $(P_i, N_i), i = 1 \ldots 50,000$ as:

$$P_i = \{s \in \text{nbest}_i | \text{NIST}(s) \geq \text{NIST}_i^* - 1\} \quad (8)$$

$$N_i = \{s \in \text{nbest}_i | \text{NIST}(s) \leq \text{NIST}_i^* - 3\} \quad (9)$$

[2] http://www.statmt.org/wmt07/

	Error rate
Baseline model	0.4720
Baseline + DLM0	0.4290
Baseline + DLM1	0.4183
Baseline + DLM2	0.4005
Baseline + DLM3	0.3803

Table 2: Error rates for fluency ranking. See article body for an explanation of the experiments.

	NIST	BLEU
Baseline model	6.9683	0.2704
Baseline + DLM0	6.9804	0.2705
Baseline + DLM1	6.9857	0.2709
Baseline + DLM2	7.0288	0.2745
Baseline + DLM3	7.0815	0.2770

Table 3: NIST and BLEU scores upon n-best list re-ranking with the proposed discriminative language models.

where NIST_i^* is the highest sentence-level NIST score achieved in nbest_i. The size of n-best lists was set to 10. Using this dataset, we trained discriminative language models by standard perceptron, confidence-weighted learning and confidence-weighted learning with soft margin.

We then trained the weights of a re-ranker using eight features (seven from the baseline Matrax plus one from the DLM) using a simple structured perceptron algorithm on the development set.

For testing, we used the same trained Matrax model to generate n-best lists of size 1,000 each for each source sentence. Then, we used the trained discriminative language model to compute a score for each translation in the n-best list. The score is used with seven standard Matrax features for re-ranking. Finally, we measure the quality of the translations re-ranked to the top.

In order to obtain the required factors for the target-side tokens, we ran the morphological analyzer and POS-tagger integrated in the Xerox Incremental Parser (XIP, Ait-Mokhtar et al. (2001)) on the target side of the training corpus used for creating the phrase-table, and extended the phrase-table format so as to record, for each token, all its factors.

5.2 Results

In the first experiment, we measure the quality of the re-ranked n-best lists by classification error rate. The error rate is computed as the fraction of pairs from a test-set which is ranked correctly according to its fluency score (approximated here by the NIST score). Results are in Table 2.

For the baseline, we use the seven default Matrax features, including a generative language model score. DLM* are discriminative language models trained using, respectively, POS features only (DLM 0) or factored features by standard perceptron (DLM 1), confidence-weighted learning (DLM 2) and confidence-weighted learning with soft margin (DLM 3). All discriminative language models strongly reduce the error rate compared to the baseline (9.1%, 11.4%, 15.1%, 19.4% relative reduction, respectively). Recall that the training set for these discriminative language models is a relatively small subset of the one used to train Matrax's integrated generative language model. Amongst the four discriminative learning algorithms, we see that factored features are slightly better then POS features, confidence-weighted learning is slightly better than perceptron, and confidence-weighted learning with soft margin is the best (9.08% and 5.04% better than perceptron and confidence-weighted learning with hard margin).

In the second experiment, we use standard NIST and BLEU scores for evaluation. Results are in Table 3. The relative quality of different methods in terms of NIST and BLEU correlates well with error rate. Again, all three discriminative language models could improve performances over the baseline. Amongst the three, confidence-weighted learning with soft margin performs best.

6 Related Work

This work is related to several existing directions: generative factored language model, discriminative language models, online passive-aggressive learning and confidence-weighted learning.

Generative factored language models are proposed by (Bilmes and Kirchhoff, 2003). In this work, factors are used to define alternative back-off paths in case surface-form n-grams are not observed a sufficient number of times in the train-

ing corpus. Unlike ours, this model cannot consider simultaneously multiple factored features coming from the same token n-gram, thus integrating all possible available information sources.

Discriminative language models have also been studied in speech recognition and statistical machine translation (Roark et al., 2007; Li and Khudanpur, 2008). An attempt to combine factored features and discriminative language modeling is presented in (Mahé and Cancedda, 2009). Unlike us, they combine together instances from multiple n-best lists, generally not comparable, in forming positive and negative instances. Also, they use an SVM to train the DLM, as opposed to the proposed online algorithms.

Our approach stems from Passive-Aggressive algorithms proposed by (Crammer et al., 2006) and the CW online algorithm proposed by (Dredze et al., 2008). In the former, Crammer et al. propose an online learning algorithm with soft margins to handle noise in training data. However, the work does not consider the confidence associated with estimated feature weights. On the other hand, the CW online algorithm in the later does not consider the case where the training data is noisy.

While developed independently, our soft-margin extension is closely related to the *AROW(project)* algorithm of (Crammer et al., 2009; Crammer and Lee, 2010). The cited work models classifiers as non-correlated Gaussian distributions over weights, while our approach uses point estimates for weights coupled with confidence scores. Despite the different conceptual modeling, though, in practice the algorithms are similar, with point estimates playing the same role as the mean vector, and our (squared) confidence score matrix the same role as the precision (inverse covariance) matrix. Unlike in the cited work, however, in our proposal, confidence scores are updated also upon correct classification of training examples, and not only on mistakes. The rationale of this is that correctly classifying an example could also increase the confidence on the current model. Thus, the update formulas are also different compared to the work cited above.

7 Conclusions

We proposed a novel approach to discriminative language models. First, we introduced the idea of using factored features in the discriminative language modeling framework. Factored features allow the language model to capture linguistic patterns at multiple levels of abstraction. Moreover, the discriminative framework is appropriate for handling highly overlapping features, which is the case of factored features. While we did not experiment with this, a natural extension consists in using all n-grams *up to* a certain order, thus providing back-off features and enabling the use of higher-order n-grams. Second, for learning factored language models discriminatively, we adopt a simple confidence-weighted algorithm, limiting the problem of poor estimation of weights for rare features. Finally, we extended confidence-weighted learning with soft margins to handle the case where labels of training data are noisy. This is typically the case in discriminative language modeling, where labels are obtained only indirectly.

Our experiments show that combining all these elements is important and achieves significant translation quality improvements already with a weak form of integration: n-best list re-ranking.

References

Salah Ait-Mokhtar, Jean-Pierre Chanod, and Claude Roux. 2001. A multi-input dependency parser. In *Proceedings of the Seventh International Workshop on Parsing Technologies*, Beijing, Cina.

Jeff A. Bilmes and Katrin Kirchhoff. 2003. Factored language models and generalized parallel backoff. In *Proceedings of HLT/NAACL*, Edmonton, Alberta, Canada.

Koby Crammer and Daniel D. Lee. 2010. Learning via gaussian herding. In *Pre-proceeding of NIPS 2010*.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *Journal Of Machine Learning Research*, 7.

Koby Crammer, Alex Kulesza, and Mark Dredze. 2009. Adaptive regularization of weight vectors. In *Advances in Neural Processing Information Systems (NIPS 2009)*.

Mark Dredze, Koby Crammer, and Fernando Pereira. 2008. Confidence-weighted linear classifiers. In *Proceedings of ICML*, Helsinki, Finland.

Zhifei Li and Sanjeev Khudanpur. 2008. Large-scale discriminative n-gram language models for statistical machine translation. In *Proceedings of AMTA*.

Pierre Mahé and Nicola Cancedda. 2009. Linguistically enriched word-sequence kernels for discriminative language modeling. In *Learning Machine Translation*, NIPS Workshop Series. MIT Press, Cambridge, Mass.

Brian Roark, Murat Saraclar, Michael Collins, and Mark Johnson. 2004. Discriminative language modeling with conditional random fields and the perceptron algorithm. In *Proceedings of the annual meeting of the Association for Computational Linguistics (ACL)*, Barcelona, Spain.

Brian Roark, Murat Saraclar, and Michael Collins. 2007. Discriminative n-gram language modeling. *Computer Speech and Language*, 21(2).

M. Simard, N. Cancedda, B. Cavestro, M. Dymetman, E. Gaussier, C. Goutte, and K. Yamada. 2005. Translating with non-contiguous phrases. In Association for Computational Linguistics, editor, *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language*, pages 755–762, October.