A NOVEL APPROACH TO PART–OF–SPEECH TAGGING BASED ON LATENT ANALOGY

Jerome R. Bellegarda

Speech & Language Technologies
Apple Inc., Cupertino, California 95014

ABSTRACT

Part-of-speech tagging is a necessary pre-processing step for many natural language tasks. Recent statistical approaches, such as conditional random fields, rely on well chosen feature functions to ensure that important characteristics of the empirical training distribution are reflected in the trained model. In practice, however, it is not always clear how to best select these feature functions in order to obtain a suitably robust model. This paper proposes an alternative strategy based on the principle of *latent analogy*. For each sentence under consideration, we construct a neighborhood of globally relevant training sentences through an appropriate data-driven mapping of the input surface form. Tagging then proceeds via locally optimal sequence alignment and maximum likelihood position scoring. Empirical evidence shows that this solution is competitive with state-of-the-art Markovian techniques.

Index Terms— Syntactic labeling, statistical tagging, POS disambiguation, global filtering, latent semantic mapping.

1. INTRODUCTION

Part-of-speech (POS) tagging is a necessary pre-processing step for many natural language processing (NLP) tasks, from text chunking to semantic role labeling. As POS tags augment the information contained within words by explicitly indicating some of the structure inherent in language, their accuracy is often critical to such downstream NLP applications.

While non-model-based solutions have also been proposed [1], in recent years tagging has traditionally involved HMMs: cf., e.g., [2], [3]. Given a suitable text corpus, HMMs can easily be trained to identify the most likely sequence of tags for the observed set of words in a given sentence. Their generative nature, however, forces them to effectively set aside valuable model parameters to account (unnecessarily, in this case) for the observation sequence.

This has sparked interest in conditional models, like maximum entropy Markov models (MEMMs), which can directly account for the conditional probability of the tag sequences given a particular observation sequence: cf., e.g., [4], [5]. MEMMs rely on a set of feature functions acting as marginal constraints to ensure that important characteristics of the empirical training distribution are reflected in the trained model. With well chosen functions covering sufficiently rich features of the training data, maximum entropy models can result in a substantially reduced tag error rate compared to HMMs [5]. Yet they can also suffer from *label bias*, whereby states with low entropy transition distributions are unduly favored [6].

Conditional random fields (CRFs) were originally introduced to overcome this weakness, and thus take full advantage of the conditional probabilistic framework [6]. These models are a form of undirected graphical models, which define a single log-linear distribution over the entire tag sequence given a particular observation sequence. This single distribution allows states to pass on any amount of probability mass to their successor states, thereby preventing label bias. As a result, CRF taggers are nominally able to outperform both HMM- and MEMM-based systems. However, this can only be achieved given adequate initial conditions, which may require an MEMM to be trained as initial starting point [6].

Hence, in practice, the tagging accuracy of both MEMMs and CRFs is essentially contingent on the specification of a high quality set of feature functions. Such selection is likely to depend on at least some measure of task-specific linguistic knowledge, complicating deployment across different applications.

The goal of this paper is to explore a completely different avenue, and design a POS tagger based on the principle of *latent analogy*. As the name implies, the inspiration for this strategy comes from an approach recently developed for the purpose of grapheme-to-phoneme conversion, dubbed *pronunciation by latent analogy* [7]. This effort itself evolved as an unconventional application of *latent semantic mapping* (LSM), a data-driven framework for modeling global relationships implicit in large volumes of data [8]. The objectives are to (i) use LSM to construct, for each sentence under consideration, a neighborhood of globally relevant training sentences, (ii) extract the associated POS sequences, and then (iii) leverage this targeted evidence in the tagging process. Interestingly, in most cases POS disambiguation seems to emerge automatically as a by-product of LSM-based semantic consistency, which *de facto* bypasses the need for any explicit linguistic knowledge.

The paper is organized as follows. The next section motivates a latent analogy approach to the problem, and Section 3 gives a general overview of the proposed framework. In Sections 4 and 5, we address the two main building blocks of tagging by latent analogy, sentence neighborhoods and sequence alignment. Finally, Section 6 reports the outcome of experimental evaluations conducted on two different corpora, one primarily for the purpose of benchmark comparisons, and the other of interest in the context of a concatenative speech synthesis task.

2. MOTIVATION

Given a natural language sentence comprising L words, the aim of POS tagging is to annotate each observed word w_i with some suitable part-of-speech p_i ($1 \leq i \leq L$). Representing the overall sequence of words by W and the corresponding sequence of POS by P, we therefore need to maximize the conditional probability $\Pr(P|W)$ over all possible POS sequences P. Maximum entropy solutions rely on log-linear models involving feature functions defined over local states in the associated graph, where each feature function expresses some selected characteristic of the empirical training distribution [6]. Due to the intrinsic lopsided sparsity of language, however, in practice many distributional aspects cannot be properly taken into account. In those specific contexts where they happen to matter, this may result in erroneous tagging.

To illustrate, consider, for example, the sentence:

Jet streams blow in the troposphere. (1)

whose correct tagging reads:

jet/NN streams/NNS blow/VBP in/IN the/DT troposphere/NN (2)

with the standard Penn Treebank POS tagset [9]. Using the CRF implementation available in [10], we obtain instead:

jet/NN streams/VBZ blow/NN in/IN the/DT troposphere/NN (3)

which incorrectly resolves the inherent POS ambiguity in the subsequence "streams blow." The problem, of course, is that from a purely syntactic viewpoint both interpretations are perfectly acceptable (a frequent situation due to the many dual noun-verb possibilities in English). What would clearly help in this case is taking into account the semantic information available. Indeed the word "troposphere," for example, would seem to make the verbal usage of "blow" quite a bit more likely.

This observation motivates an LSM approach to the problem, so dimensionality reduction can be leveraged to extract global information about the sentence to be tagged. LSM has already proven effective over the past two decades in a variety of other fields, including query-based information retrieval, word clustering, document/topic clustering, large vocabulary language modeling, and semantic inference for voice command and control [8]. In the present case, LSM is used at the sentence level, relying on co-occurrence relationships in the training corpus in order to map each training sentence onto an appropriate vector space. Once this is done, the projection of the current sentence onto that space can then be exploited to inform the construction of the desired POS sequence. This strategy, although very different from conditional Markov approaches, adheres to the same general principle of using empirical evidence as a constraint in parameter estimation.

3. TAGGING BY LATENT ANALOGY

The associated framework is illustrated in Fig. 1. During the training phase, a global LSM analysis is performed on the available corpus. This leads to a representation of each training sentence in terms of a *sentence anchor* in a suitable feature space. Each sentence anchor can then be associated with its corresponding POS sequence from the labeled training corpus. During the tagging phase, the same LSM analysis is performed on the input sentence, with the goal of determining which sentences from the training data are most related to it. This leads to the concept of *sentence neighborhood*. Loosely speaking, two sentences belong to the same neighborhood if they share the same underlying subject matter (as modeled by LSM). Note that this notion of neighborhood is markedly different from the one in [11], where the goal was to collect implicit negative evidence about the overall syntax of the sentence. Here the objective is to assess the semantic "closeness" between the input and each training sentence. If a training sentence is deemed "similar" enough, it is added to the sentence neighborhood of the input sentence.

Thus, neighborhood construction can be viewed as a mechanism to zero in on "relevant" (global) features of the training data, which is somewhat analogous to selecting "important" distributional aspects in MEMM-CRF. In that sense, LSM plays a role comparable to that of feature selection in the conventional framework. In fact, it is clear that LSM shares a major advantage of feature functions (compared to standard HMM), namely the ability to integrate long-distance dependencies between observation elements (albeit in the form of global co-occurrences, instead of non-independent overlapping characteristics of the training distribution). However, here this integration happens automatically as a result of (data-driven) neighborhood construction, rather than as a consequence of injecting external (and likely task-dependent) linguistic knowledge. In addition, the notion of "closeness" is defined across the entire corpus, so features with higher relevance across multiple contexts can be expected to contribute more prominently.

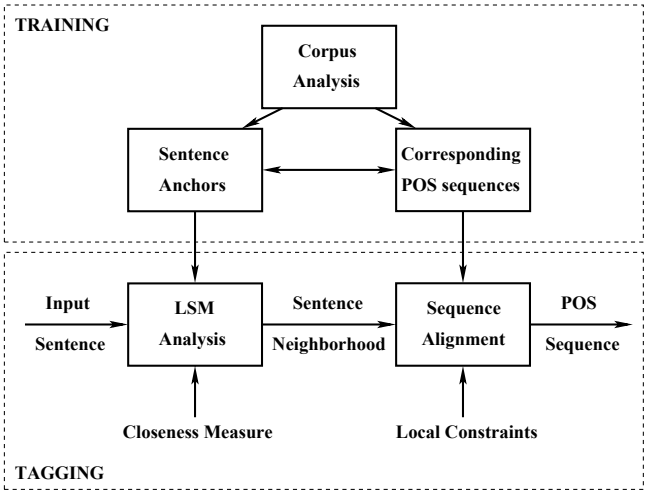

Fig. 1. *POS Tagging by Latent Analogy Framework.*

Once a sentence neighborhood is specified for a given input sentence, the corresponding POS sequences are extracted accordingly. By construction, these sequences have the property that they contain at least one sub-sequence which is "locally close" to the POS sequence sought. Assuming that the global properties of the sentence implicitly entail POS consistency at that location, aligning these subsequences allows us to expose promising common elements between them. The more common a particular POS in a particular position, the more likely it is to correspond to a correct tag. The maximum likelihood estimate at every position is therefore the best candidate for the final POS sequence. It remains to proceed in a left-to-right fashion, using words in the input sentence as sequential "landmarks" acting like local constraints, to let the final tags emerge spontaneously from the alignment process.

The procedure of Fig. 1 is entirely data-driven and requires no human supervision (beyond the original annotation of the training corpus). Compared to MEMM-CRF, it essentially decouples feature selection from POS sequence assembly. Now neighborhood generation involves gathering globally pertinent information on the observation side, while final assembly involves exploiting locally consistent constraints on the POS side. This is particularly effective when it comes to disambiguating alternative POS sub-sequences on the basis of semantic usage.

4. SENTENCE NEIGHBORHOODS

Let $\mathcal{T}$, $|\mathcal{T}| = N$, be a collection of training sentences, where each word has been annotated with its corresponding POS, and $\mathcal{V}$, $|\mathcal{V}| = M$, the associated set of all n-grams observed in the collection (i.e., the underlying vocabulary if $n = 1$), including proper markers for punctuation, etc. The LSM paradigm defines a mapping between the discrete sets $\mathcal{V}$, $\mathcal{T}$ and a continuous vector space $\mathcal{L}$, whereby each

Table I. Sentence Neighborhood for Sentence (1).

jet/*NN propulsion/NN also/RB makes/VBZ flight/NN possible/JJ at/IN extremely/RB high/JJ altitudes/NNS ,/, and/CC even/RB in/IN outer/JJ space/NN*
these/DT superalloys/NNS are/VBP important/JJ components/NNS of/IN **jet**/*NN engines/NNS and/CC spacecraft/NN*
high-speed/JJ **streams**/*NNS of/IN the/DT solar/JJ wind/NN appear/VBP as/IN the/DT sun/NN 's/POS activity/NN increases/NNS*
this/DT device/NN sprays/VBZ **streams**/*NNS of/IN vapor/NN that/WDT sweep/VBP gas/NN molecules/NNS out/IN of/IN the/DT enclosed/VBN space/NN*
grade/NN separations/NNS are/VBP often/RB used/VBN to/TO separate/VB crossing/VBG **streams**/*NNS of/IN traffic/NN*
extremely/RB strong/JJ winds/NNS **blow**/*VBP in/IN this/DT layer/NN*
westerlies/NNP and/CC trade/NN winds/NNS **blow**/*VBP away/RB from/IN the/DT thirty/CD degrees/NNS latitude/VBP belt/NN*
similar/JJ winds/NNS that/WDT **blow**/*VBP in/IN other/JJ parts/NNS of/IN the/DT world/NN are/VBP called/VBN foehns/NNS*
the/DT temperature/NN in/IN a/DT thin/JJ layer/NN of/IN the/DT **troposphere**/*NN then/RB increases/VBZ with/IN altitude/NN*
other/JJ parts/NNS of/IN the/DT atmosphere/NN are/VBP above/IN the/DT **troposphere**/*NN*
most/JJS clouds/NNS occur/VBP within/IN the/DT **troposphere**/*NN*

n-gram in $\mathcal{V}$ is represented by a vector $\bar{u}_i$ in $\mathcal{L}$, and each training sentence in $\mathcal{T}$ is represented by a vector $\bar{v}_j$ in $\mathcal{L}$. The vector space $\mathcal{L}$ is known as the *latent semantic space* associated with the training collection.

We follow the established LSM mechanisms for deriving this vector space $\mathcal{L}$, as well as mapping the input sentence to it. For the sake of brevity, the reader is referred to [12] for the details of this procedure. The steps involved are: (i) constructing the $(M \times N)$ co-occurrence matrix W, with entries which suitably reflect the extent to which each n-gram in $\mathcal{V}$ appeared in each sentence in $\mathcal{T}$; (ii) performing a singular value decomposition (SVD) of W, keeping only the R leading singular values; (iii) augmenting the matrix W to find the proper representation of a given input sentence in the resulting vector space of dimension R; and (iv) defining a suitable closeness measure on this R-dimensional feature space [12].

Using this closeness measure, it is then a simple matter to rank all training sentences in decreasing order of closeness to the representation of the input sentence. The associated sentence neighborhood follows by retaining only those instances whose closeness measure is higher than a pre-set threshold. To illustrate, an actual (partial) sentence neighborhood for the example selected in Section 2 is reported in Table I. For reasons to become clear shortly, in this case we have ordered the (adequately tagged) sentences separately for each reference word (in bold).

5. SEQUENCE ALIGNMENT

Since the sentence neighborhood thus constructed is made up of training sentences, associated POS sequences are readily available from the labeled training corpus. In principle, each of these POS sequences contains at least one sub-sequence which is germane to the input sentence. Thus, the final POS sequence can be assembled by judicious alignment of appropriate POS sub-sequences from the sentence neighborhood.

In pronunciation by latent analogy, a similar alignment problem is solved by using a sequence analysis approach commonly used in molecular biology [7]. But, in this application, the relevant phoneme sub-strings have to emerge from the alignment itself. Here, the problem is comparatively simpler, because each POS value is attached to its own word, whose identity is known. As a result, it is only necessary to align sub-sequences in the vicinity of each word, as opposed to the added complexity of aligning complete POS sequences.

We therefore proceed word by word, collecting at each step the POS sub-sequences from entries in the sentence neighborhood containing the relevant reference word. In accordance to the remark above, we only retain $(2K + 1)$ POS in each sub-sequence, centered around that of the current word. Proceeding left-to-right, we thus obtain a set of POS values for each word, where each value is presumably consistent with global information extracted from the training corpus and germane to the input sentence. The maximum likelihood estimate is then computed for every word, by simply using the observed POS counts at this position. The outcome is the final POS sequence sought.

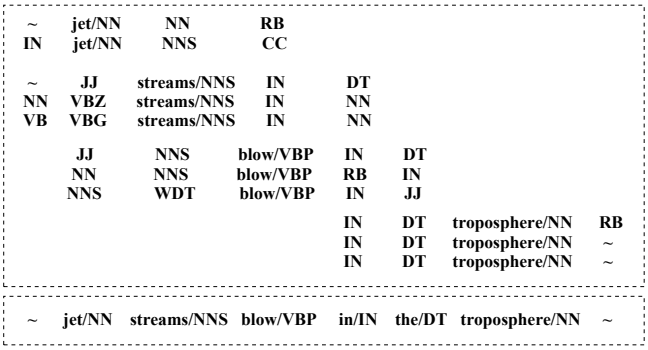

Fig. 2. *Example of Sequence Alignment for (1).*

This process is illustrated in Fig. 2 for the sentence (1) of Section 2. Given the sentence neighborhood listed in Table I, we proceed left to right in the order of each reference word to obtain the alignment presented in the top box. Note that in this example we retain POS sub-sequences using a local scope of size $K = 2$. Maximum likelihood POS assembly then leads to the final POS sequence given in (2). In this case, tagging by latent analogy is able to satisfactorily resolve the inherent POS ambiguity discussed previously.

6. EXPERIMENTS

As pointed out in [3], most tagging accuracies reported in the literature are not directly comparable, because of different tagsets, different test corpora, or possibly different randomized and irreproducible splits of training and test data. The closest thing to a canonical setup is the training, development, and test split of the Penn Treebank described in [2], which was also used in [3] and [5] for testing their HMM and MEMM taggers, respectively.

Table II. Results on Penn Treebank.

Tagging Approach	Tag Error Rate
Baseline (Frequency)	7.8 %
Standard HMM	4.1 %
Contextualized HMM	3.4 %
Feature-Rich MEMM	2.8 %
CRF	2.8 %
Tagging by Latent Analogy	3.5 %

We thus followed [3] in the allocation of sections 00-18 for training, 19-21 for development, and 22-24 for testing, as well as other aspects of the experimental setup, such as OOV elimination and lexicon filtering. As baseline we also used the same tagger as [3], i.e., the tagger which always chooses a word's most frequent tag regardless of context. We then compared the CRF implementation in Section 2 with tagging by latent analogy as described in Sections 4–5.

In the latter implementation we used $n = 1$ (unigrams only), which led to values of M and N on the order of $30,000$ and $40,000$, respectively. This corresponds to a (sparse) matrix W of moderate size, for which the SVD can be done efficiently.[1] Sentence anchors were obtained using $R = 100$ for the order of the decomposition, and on the average each sentence neighborhood comprised about 50 entries. Sequence alignment proceeded using a local scope of $K = 2$ POS on each side, and the final POS sequence was produced using the maximum likelihood estimate at each position, with tag frequency used as tie-breaker, as necessary.

The results are summarized in Table II, which also recalls the results obtained on the same test set with a standard HMM model, the contextualized HMM model of [3], and the carefully tuned feature-rich MEMM approach of [5], as reported in [3].

It can be seen that the performance of tagging by latent analogy is roughly comparable to that of contextualized HMM, but substantially inferior to MEMM-CRF. At this point we wondered whether this outcome might somehow be linked to the fairly homogeneous writing style characteristic of the data. To get a more precise idea of tagging robustness, we considered another test set, extracted from an internal corpus used to record speech segments for concatenative speech synthesis (cf. [13]). This corpus, by construction, contains more diverse material assembled from a greater variety of sources, and in particular includes a greater proportion of text written in a more casual, spoken style (as can be found in blogs, for instance). The results, on a test set of approximately 2500 sentences, are summarized in Table III.

This time the performance of tagging by latent analogy is roughly the same as that of CRF. In fact, a comparison between Table II and Table III shows that latent analogy seems largely unaffected by the underlying changes, while CRF performance degrades by about 26%. This suggests that tagging by latent analogy is more robust across a greater variety of different syntactic contexts, whereas in order to optimally reflect the new material the feature functions in MEMM-CRF would probably need to be appropriately fine-tuned. This remark is consistent with our initial conjecture that global information useful to POS disambiguation can be captured more systematically (and expediently) with LSM than via conventional feature selection.

[1] We performed the SVD using the single vector Lanczos method as done in [12]. To fix ideas, on a 2.33 GHz Intel Core 2 Duo CPU, this took less than a minute of CPU time.

Table III. Results on Speech Synthesis Corpus.

Tagging Approach	Tag Error Rate
Baseline (Frequency)	9.9 %
Tagging by Latent Analogy	3.6 %
CRF	3.4 %

7. CONCLUSION

We have proposed an alternative strategy for POS tagging, adapted from pronunciation by latent analogy, which focuses on two loosely coupled sub-problems: (i) extract from the training corpus those sentences which are the most germane in a global sense, and (ii) exploit the evidence thus gathered to assemble the POS sequence based on local constraints. We address (i) by leveraging the latent topicality of every sentence, as uncovered by a global LSM analysis of the entire training corpus. Each input surface form thus leads to its own customized neighborhood, comprising those training sentences which are most related to it. POS tagging then follows via locally optimal sequence alignment and maximum likelihood position scoring, in which the influence of the entire neighborhood is implicitly and automatically taken into account. This method was observed to be effective on two different corpora: a subset of the Penn Treebank suitable for conducting benchmark comparisons, and an internal corpus of more diverse material used in a concatenative speech synthesis task. In practice, tagging by latent analogy is likely to achieve close to the same level of performance as maximum entropy tagging, at a fraction of the cost. This bodes well for its general deployability across a wide range of applications.

8. REFERENCES

[1] E. Brill, "Unsupervised Learning of Disambiguation Rules for Part of Speech Tagging," in *Proc. 3rd Workshop Very Large Corpora*, Boston, MA, pp. 1-13, 1995.

[2] M. Collins, "Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms," in *Proc. Conf. Empirical Methods Natural Language Proc.*, Philadelphia, PA, pp. 1–8, July 2002.

[3] M. Banko and R.C. Moore, "Part of Speech Tagging in Context," in *Proc. 20th Int. Conf. Computational Linguistics (COLING'04)*, Geneva, Switzerland, pp. 556–561, August 2004.

[4] S. Della Pietra, V. Della Pietra, and J. Lafferty, "Inducing Features of Random Fields," *IEEE Trans. Pattern Analysis Machine Intelligence*, Vol. 19, No. 4, pp. 380–393, 1997.

[5] K. Toutanova, D. Klein, C. Manning, and Y. Singer, "Feature–Rich Part–of–Speech Tagging with a Cyclic Dependency Network," in *Proc. HLT–NAACL*, Edmonton, Canada, pp. 252–259, May 2003.

[6] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", in *Proc. 18th Int. Conf. Machine Learning (ICML 2001)*, Williamstown, MA, pp. 282–289, June 2001.

[7] J.R. Bellegarda, "Unsupervised, Language-Independent Grapheme-to-Phoneme Conversion by Latent Analogy," *Speech Communication*, Vol. 46, No. 2, pp. 140-152, Amsterdam, The Netherlands: Elsevier Science, June 2005.

[8] J.R. Bellegarda, "Latent Semantic Mapping: A Data–Driven Framework for Modeling Global Relationships Implicit in Large Volumes of Data," *IEEE Signal Processing Magazine*, Vol. 22, No. 5, pp. 70–80, September 2005.

[9] M.P. Marcus, B. Santorini, and M.A. Marcinkiewicz, "Building a Large Annotated Corpus of English: the Penn Treebank," *Computational Linguistics*, Vol. 19, No. 2, pp. 313-330, 1993.

[10] S. Sarawagi, "CRF Package for Java," `http://crf.sourceforge.net/`, 2004.

[11] N.A. Smith and J. Eisner, "Contrastive Estimation: Training Log–Linear Models on Unlabeled Data," in *Proc. 43rd ACL*, Ann Arbor, MI, pp. 354–362, June 2005.

[12] J.R. Bellegarda, "Exploiting Latent Semantic Information in Statistical Language Modeling," *Proc. IEEE*, Vol. 88, No. 8, pp. 1279–1296, August 2000.

[13] J.R. Bellegarda, K.E.A. Silverman, K.A. Lenzo, and V. Anderson, "Statistical Prosodic Modeling: From Corpus Design to Parameter Estimation," *IEEE Trans. Speech Audio Proc.*, Vol. SAP–9, No. 1, pp. 52–66, January 2001.