Using the Distribution of Performance for Studying Statistical NLP Systems and Corpora

Yuval Krymolowski
Department of Mathematics and Computer Science
Bar-Ilan University
52900 Ramat Gan, Israel

Abstract

Statistical NLP systems are frequently evaluated and compared on the basis of their performances on a single split of training and test data. Results obtained using a single split are, however, subject to sampling noise. In this paper we argue in favour of reporting a distribution of performance figures, obtained by resampling the training data, rather than a single number. The additional information from distributions can be used to make statistically quantified statements about differences across parameter settings, systems, and corpora.

1 Introduction

The common practice in evaluating statistical NLP systems is using a standard corpus (e.g., Penn TreeBank for parsing, Reuters for text categorization) along with a standard split between training and test data. As systems improve, it becomes harder to achieve additional improvements, and the performance of various state-of-the-art systems is approximately identical. This makes performance comparisons difficult.

In this paper, we argue in favour of studying the *distribution* of performance, and present conclusions drawn from studying the recall distribution. This distribution provides measures for answering the following questions:

Q1: Comparing systems on given data: Is classifier A better than classifier B for given training and test data?

Q2: Adequacy of training data to test data: Is a system trained on dataset X adequate for analysing dataset Y? Are features from X indicative in Y?

Q3: Comparing data sets with a given system: If a different training set improves the result of system A on dataset Y_1, will this be the case on dataset Y_2 as well?

The answers to these questions can provide useful insight into statistical NLP systems. In particular, about sensitivity to features in the training data, and transferability. These properties can be different even when similar performance is reported.

A statistical treatment of Question 1 is presented by Yeh (2000). He tests for the significance of performance differences on fixed training and test data sets. In other related works, Martin and Hirschberg (1996) provides an overview of significance tests of error differences in small samples, and Dietterich (1998) discusses results of a number of tests.

Questions 2 and 3 have been frequently raised in NLP, but not explicitly addressed, since the prevailing evaluation methods provide no means of addressing them. In this paper we propose addressing all three questions with a single experimental methodology, which uses the distribution of recall.

2 Motivation

Words, parts-of-speech (POS), words, or any feature in text may be regarded as outcomes

of a statistical process. Therefore, word counts, count ratios, and other data used in creating statistical NLP models are statistical quantities as well, and as such prone to *sampling noise*. Sampling noise results from the finiteness of the data, and the particular choice of training and test data.

A model is an approximation or a more abstract representation of training data. One may look at a model as a collection of estimators analogous, e.g., to the slope calculated by linear regression. These estimators are statistics with a distribution related to the way they were obtained, which may be very complicated. The performance figures, being dependent on these estimators, have a distribution function which may be difficult to find theoretically. This distribution gives rise to *intrinsic noise*.

Performance comparisons based on a single run or a few runs do not take these noises into account. Because we cannot assign the resulting statements a confidence measure, they are more qualitative than quantitative. The degree to which we can accept such statements depends on the noise level and more generally, on the distribution of performance.

In this paper, we use recall as a performance measure (cf. Section 4.4 and Section 3.2 in (Yeh, 2000)). Recall samples are obtained by resampling from training data and training classifiers on these samples.

The resampling methods used here are cross-validation and bootstrap (Efron and Gong, 1983; Efron and Tibshirani, 1993, cf. Section 3). Section 4 presents the experimental goals and setup. Results are presented and discussed in Section 5, and a summary is provided in Section 6.

3 The Bootstrap Method

The bootstrap is a re-sampling technique designed for obtaining empirical distributions of estimators. It can be thought of as a smoothed version of k-fold cross-validation (CV). The method has been applied to decision tree and bayesian classifiers by Kohavi (1995) and to neural networks by, e.g., LeBaron and Weigend (1998).

In this paper, we use the bootstrap method to obtain the distribution of performance of a system which learns to identify non-recursive noun-phrases (base-NPs). While there are a few refinements of the method, the intention of this paper is to present the benefits of obtaining distributions, rather than optimising bias or variance. We do not aim to study the properties of bootstrap estimation.

Let a statistic $S = S(x_1, \ldots, x_n)$ be a function of the independent observations $\{x_i\}_{i=1}^n$ of a statistical variable X. The bootstrap method constructs the distribution function of S by successively re-sampling x with replacements.

After B samples, we have a set of *bootstrap samples* $\{x_1^b, \ldots, x_n^b\}_{b=1}^B$, each of which yields an estimate $\hat{S}^b$ for S. The distribution of $\hat{S}$ is the bootstrap estimate for the distribution of S. That distribution is mostly used for estimating the standard deviation, bias, or confidence interval of S.

In the present work, x_i are the base-NP instances in a given corpus, and the statistic S is the recall on a test set.

4 Experimental Setup

The aim of our experiments is to test whether the recall distribution can be helpful in answering the questions Q1–Q3 mentioned in the introduction of this paper.

The data and learning algorithms are presented in Sections 4.1 and 4.2. Section 4.3 describes the sampling method in detail. Section 4.4 motivates the use of recall and describes the experiments.

4.1 Data

We used Penn-Treebank (Marcus et al., 1993) data, presented in Table 1. Wall-Street Journal (WSJ) Sections 15-18 and 20 were used by Ramshaw and Marcus (1995) as training and test data respectively for evaluating their base-NP chunker. These data have since become a standard for evaluating base-NP systems.

The WSJ texts are economic newspaper reports, which often include elaborated sentences containing about six base-NPs on the

Source	Sentences	Words	Base NPs
WSJ 15-18	8936	229598	54760
WSJ 20	2012	51401	12335
ATIS	190	2046	613
WSJ 20a	100	2479	614
WSJ 20b	93	2661	619

Table 1: Data sources

average.

The ATIS data, on the other hand, are a collection of customer requests related to flight schedules. These typically include short sentences which contain only three base-NPs on the average. For example:

```
I have a friend living in Denver
that would like to visit me
here in Washington DC .
```

The structure of sentences in the ATIS data differs significantly from that in the WSJ data. We expect this difference to be reflected in the recall of systems tested on both data sets.

The small size of the ATIS data can influence the results as well. To distinguish the size effect from the structural differences, we drew two equally small samples from WSJ Section 20. These samples, WSJ20a and WSJ20b, consist of the first 100 and the following 93 sentences respectively. There is a slight difference in size because sentences were kept complete, as explained Section 4.3.

4.2 Learning Algorithms

We evaluated base-NP learning systems based on two algorithms: MBSL (Argamon et al., 1999) and SNoW (Muñoz et al., 1999).

MBSL is a memory-based system which records, for each POS sequence containing a border (left, right, or both) of a base-NP, the number of times it appears with that border vs. the number of times it appears without it. It is possible to set an upper limit on the length of the POS sequences.

Given a sentence, represented by a sequence of POS tags, the system examines each subsequence for being a base-NP. This is done by attempting to tile it using POS sequences that appeared in the training data with the base-NP borders at the same locations.

For the purpose of the present work, suffice it to mention that one of the parameters is the *context size*(c). It denotes the maximal number of words considered before or after a base-NP when recording sub-sequences containing a border.

SNoW (Roth, 1998, "Sparse Network of Winnow") is a network architecture of Winnow classifiers (Littlestone, 1988). Winnow is a mistake-driven algorithm for learning a linear separator, in which feature weights are updated by multiplication. The Winnow algorithm is known for being able to learn well even in the presence of many noisy features.

The features consist of one to four consecutive POSs in a 3-word window around each POS. Each word is classified as a beginning of a base-NP, as an end, or neither.

4.3 Sampling Method

In generating the training samples we sampled *complete* sentences. In MBSL, an un-marked boundary may be counted as a negative example for the POS-subsequences which contains it. Therefore, sampling only part of the base-NPs in a sentence will generate negative examples.

For SNoW, each word is an example, but most of the words are neither a beginning nor an end of a base-NP. Random sampling of words might generate a sample with an improper balance between the three classes.

To avoid these problems, we sampled full sentences instead of words or instances. Within a good approximation, it can be assumed that base-NP patterns in a sentence do not correlate. The base-NP instances drawn from the sampled sentences can therefore be regarded as independent.

As described at the end of Sec. 4.1, the WSJ20a and WSJ20b data were created so that they contain 613 instances, like the ATIS data. In practice, the number of instances exceeds 613 slightly due to the full-sentence constraint. For the purpose of this work, it is enough that their size is very close to the size of ATIS.

Dataset	Sentences	Base-NPs
Training	8938 ± 48	54763 ± 2
Unique:	5648 ± 34	

Table 2: Sentence and instant counts for the bootstrap samples. The second line refers to unique sentences in the training data.

We used the WSJ15-18 dataset for training. This dataset contains $n_0 = 54760$ base-NP instances. The number of instances in a bootstrap sample depends on the number of instances in the last sampled sentence. As Table 2 shows, it is slightly more than n_0.

For k-CV sampling, the data were divided into k random distinct parts, each containing $\frac{n_0}{k} \pm 2$ instances. Table 3 shows the number of recall samples in each experiment (MBSL and SNoW experiments were carried out seperately).

Method	MBSL	SNoW
Bootstrap	2200	1000
CV (total folds)	1500	1000

Table 3: Number of bootstrap samples and total CV folds.

4.4 Experiments

We trained SNoW and MBSL; the latter using context sizes of c=1 and c=3. Data sets WSJ20, ATIS, WSJ20a, and WSJ20b were used for testing. MBSL runs with the two c values were conducted on the same training samples, therefore it is possible to compare their results directly.

Each run yielded recall and precision. Recall may be viewed as the expected 0-1 loss-function on the *given* test sample of instances. Precision, on the other hand, may be viewed as the expected 0-1 loss on the sample of instances *detected* by the learning system. Care should be taken when discussing the distribution of precision values because this sample varies from run to run. We will therefore only analyse the distribution of recall in this work.

In the following, r^1 and r^3 denote recall samples of MBSL with $c = 1$ and $c = 3$, with standard deviations σ^1 and σ^3. ρ^{13} denotes the cross-correlation between r^1 and r^3. SNoW recall and standard deviation will be denoted by r^{SN} and σ^{SN}.

To approach the questions raised in the introduction we made the following measurements:

Q1: System comparison was addressed by comparing r^1 and r^3 on the same test data. With samples at hand, we obtained an estimate of $P(r^3 > r^1)$.

Q2: We studied training and test adequacy through the effect of more specific features on recall, and on its standard deviation.

Setting $c = 3$ takes into account sequences with context of two and three words in addition to those with $c = 1$. Sequences with larger context are more specific, and an improvement in recall implies that they are informative in the test data as well.

For particular choices of parameters and test data, the recall spread yields an estimate of the training sampling noise. On inadequate data, where the statistics differ significantly from those in the training data, even small changes in the model can lead to a noticeable difference in recall. This is because the model relies on statistics which appear relatively rarely in the test data. Not only do these statistics provide little information about the problem, but even small differences in weighting them are relatively influential.

Therefore, the more training and test data differ from each other, the more spread we can expect in results.

Q3: For comparing test data sets with a system, we used cross-correlations between r^1, r^3, or r^{SN} samples obtained on these data sets. We know that WSJ data are different from ATIS data, and so expect the results on WSJ to correlate with ATIS results less than with other WSJ results.

5 Results and Discussion

For each of the five test datasets, Table 4 reports averages and standard deviations of r^1, r^3, and r^{SN} obtained by 3, 5, 10, and 20-fold cross-validation, and by bootstrap. ρ^{13} and $P(r^3 > r^1)$ are reported as well.

We discuss our results by considering to

what extent they provide information for answering the three questions:

Q1 – Comparing systems on given data: For the WSJ data sets, the difference between r^3 and r^1 was well above their standard deviations, and $r^3 > r^1$ nearly always. For ATIS, the standard deviation of the difference ($\sigma^2_{r^3-r^1} = (\sigma^1)^2 + (\sigma^3)^2 - 2\sigma^1\sigma^3 \cdot \rho^{13}$) was small due to the high ρ^{13}, and $r^1 > r^3$ nearly always.

Q2 – The adequacy of training and test sets: It is clear that adding more specific features, by increasing the context, improved recall on the WSJ test data and degraded it on the ATIS data. This is likely to be an indication of the difference in syntactic structure between ATIS and WSJ texts.

Another evidence of structural difference comes from standard deviations. The spread of the ATIS results always exceeded that of the WSJ results, with *all three experiments*. That difference cannot be solely attributed to the small size of ATIS, since WSJ20a and WSJ20b results displayed a much smaller spread. Indeed, these results had a wider standard deviation than WSJ20, probably due to the smaller size, but not as wide as ATIS. This indicates that base-NPs in ATIS text have different characteristics than those in WSJ texts.

Q3 – Comparing datasets by a system: Table 5 reports, for each pair of datasets, the correlation between the 5-fold CV recall samples of each experiment on these datasets. The correlations change with CV fold number, 5-fold results were chosen as they represent intermediary values.

Both MBSL experiments yielded negligible correlations of ATIS results with *any* WSJ data set, whether large or small. These correlations were always weaker than with WSJ20a and WSJ20b, which are about the same size.

This is due to ATIS being a different kind of text. The correlation between WSJ20a and WSJ20b results was also weak. This may be due to their small sizes; these texts might not share enough features to make a significant correlation.

SNoW results were highly correlated for all pairs. That behaviour is markedly different from the MBSL results, and indicates a high level of noise in the SNoW features. Indeed, Winnow is able to learn well in the presence of noise, but that noise causes the high correlations observed here.

5.1 Further Observations

The decrease of ρ^{13} with CV fold number is related to stabilization of the system. As the folds become larger, training samples become more similar to each other, and the spread of results decreases. This effect was not visible in the SNoW data, most likely due to the high level of noise in the features. This noise also contributes to the higher standard deviation of SNoW results.

6 Summary and Further Research

In this work, we used the distribution of recall to address questions concerning base-NP learning systems and corpora. Two of these questions, of training and test adequacy, and of comparing data sets using NLP systems, were not addressed before.

The recall distributions were obtained using CV and bootstrap resampling.

We found differences between algorithms with similar recall, related to the features they use.

We demonstrated that using an inadequate test set may lead to noisy performance results. This effect was observed with two different learning algorithms. We also reported a case when changing a parameter of a learning algorithm improved results on one dataset but degraded results on another.

We used classifiers as "similarity rulers", for producing a similarity measure between datasets. Classifiers may have various properties as similarity rulers, even when their recalls are similar. Each classifier should be scaled differently according to its noise level. This demonstrates the way we can use classifiers to study data, as well as use data to study classifiers.

Test data	Method	MBSL				SNoW
		$E(r^1) \pm \sigma^1$	$E(r^3) \pm \sigma^3$	ρ^{13}	$P(r^3 > r^1)$	$E(r^{\mathrm{SN}}) \pm \sigma^{\mathrm{SN}}$
WSJ 20	3-CV	89.64 ± 0.16	91.26 ± 0.12	0.36	100%	90.18 ± 1.01
	5-CV	89.75 ± 0.14	91.43 ± 0.10	0.30	100%	90.37 ± 1.03
	10-CV	89.80 ± 0.12	91.53 ± 0.08	0.25	100%	90.47 ± 1.11
	20-CV	89.81 ± 0.11	91.56 ± 0.07	0.28	100%	90.51 ± 1.19
	Bootstrap	89.58 ± 0.17	91.16 ± 0.14	0.42	100%	89.83 ± 0.93
	$E(\cdot)$	89.74	91.58			91.23
ATIS	3-CV	85.70 ± 2.03	83.99 ± 1.87	0.82	3%	83.70 ± 4.11
	5-CV	85.76 ± 1.87	83.69 ± 1.57	0.79	1%	83.53 ± 4.52
	10-CV	85.90 ± 1.31	84.78 ± 0.92	0.78	4%	83.38 ± 5.14
	20-CV	85.78 ± 1.16	83.28 ± 0.85	0.77	0%	83.23 ± 5.36
	Bootstrap	85.72 ± 1.95	84.69 ± 1.95	0.81	16%	83.50 ± 3.35
	$E(\cdot)$	85.81	83.20			85.48
WSJ 20a	3-CV	89.45 ± 0.42	91.25 ± 0.56	0.33	100%	90.84 ± 1.04
	5-CV	89.66 ± 0.36	91.64 ± 0.54	0.32	100%	91.07 ± 1.15
	10-CV	89.79 ± 0.28	91.85 ± 0.49	0.20	100%	91.14 ± 1.26
	20-CV	89.82 ± 0.23	91.89 ± 0.44	0.18	100%	91.11 ± 1.39
	Bootstrap	89.42 ± 0.47	91.55 ± 0.57	0.33	99%	90.76 ± 1.00
	$E(\cdot)$	89.73	92.18			90.07
WSJ 20b	3-CV	88.95 ± 0.41	90.12 ± 0.39	0.37	99%	89.79 ± 0.81
	5-CV	89.03 ± 0.36	90.15 ± 0.31	0.31	99%	89.81 ± 0.84
	10-CV	89.06 ± 0.33	90.14 ± 0.22	0.28	99%	89.83 ± 0.86
	20-CV	89.07 ± 0.27	90.13 ± 0.18	0.22	100%	89.87 ± 0.88
	Bootstrap	89.00 ± 0.44	90.17 ± 0.44	0.38	98%	89.93 ± 0.80
	$E(\cdot)$	89.01	91.55			90.79

Table 4: Recall statistic summary for MBSL with contexts $c = 1$ and $c = 3$, and SNoW. The $E(\cdot)$ figures were obtained using the full training set. Note the monotonic change of standard deviation with fold number. The s.d. of the bootstrap samples are closest to those of low-fold CV samples.

5-CV	WSJ 20b			WSJ 20a			ATIS		
	r^1	r^3	r^{SN}	r^1	r^3	r^{SN}	r^1	r^3	r^{SN}
WSJ 20	0.33	0.19	0.72	0.26	0.29	0.78	0.08	0.02	0.76
ATIS	-0.01	0.00	0.59	0.02	-0.01	0.63			
WSJ 20a	0.07	0.04	0.59						

Table 5: Cross-correlations between recalls of the three experiments on the test data for five-fold CV. Correlations of r^1 capture dataset similarity in the best way.

By using MBSL with different context sizes, our results provide insights into the relation between training and test data sets, in terms of general and specific features. That issue becomes important when one plans to use a system trained on certain data set for analysing an arbitrary text. Another approach to this topic, examining the effect of using lexical bigram information, which is very corpus-specific, appears in (Gildea, 2001).

In our experiments with systems trained on WSJ data, there was a clear difference between their behaviour on other WSJ data and on the ATIS data set, in which the structure of base-NPs is different. That difference was observed with correlations and standard deviations. This shows that resampling the training data is essential for noticing these structure differences.

To control the effect of small size of the ATIS dataset, we provided two equally-small WSJ data sets. The effect of different genres was stronger than that of the small-size.

In future study, it would be helpful to study the distribution of recall using training and test data from a few genres, across genres, and on combinations (e.g. "known-similarity corpora" (Kilgarriff and Rose, 1998)). This will provide a measure of the transferability of a model.

We would like to study whether there is a relation between bootstrap and 2 or 3-CV results. The average number of unique base-NPs in a random bootstrap training sample is about 63% of the total training instances (Table 2). That corresponds roughly to the size of a 3-CV training sample. More work is required to see whether this relation between bootstrap and low-fold CV is meaningful.

We also plan to study the distribution of precision. As mentioned in Sec. 4.4, the precisions of different runs are now taken from different sample spaces. This makes the bootstrap estimator unsuitable, and more study is required to overcome this problem.

References

S. Argamon, I. Dagan, and Y. Krymolowski. 1999. A memory-based approach to learning shallow natural language patterns. *Journal of Experimental and Theoretical AI*, 11:369–390. CMP-LG/9806011.

T. G. Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7).

Bradley Efron and Gail Gong. 1983. A leisurely look at the bootstrap, the jackknife, and cross-validation. *Am. Stat.*, 37(1):36–48.

Bradley Efron and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman and Hall, New York.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proc. 2001 Conf. on Empirical Methods in Natural Language Processing (EMNLP–2001)*, Carnegie Mellon University, Pittsburgh, June. ACL-SIGDAT.

Adam Kilgarriff and Tony Rose. 1998. Measures for corpus similarity and homogeneity. In *Proc. 3rd Conf. on Empirical Methods in Natural Language Processing (EMNLP–3)*, pages 46–52, Granada, Spain, June. ACL-SIGDAT.

Ron Kohavi. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *proceedings of the International Joint Conference on Artificial Intelligence*, pages 1137–1145.

B. LeBaron and A. S. Weigend. 1998. A bootstrap evaluation of the effect of data splitting on financial time series. *IEEE Transactions on Neural Networks*, 9(1):213–220, January.

N. Littlestone. 1988. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2:285–318.

M. P. Marcus, B. Santorini, and M. Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, June.

J. Martin and D. Hirschberg. 1996. Small sample statistics for classification error rates II: Confidence intervals and significance tests. Technical report, Dept. of Information and Computer Science, University of California, Irvine. Technical Report 96-22.

M. Muñoz, V. Punyakanok, D. Roth, and D. Zimak. 1999. A learning approach to shallow parsing. In *EMNLP-VLC'99, the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 168–178, June.

L. A. Ramshaw and M. P. Marcus. 1995. Text chunking using transformation-based learning. In *Proceedings of the Third Workshop on Very Large Corpora*.

D. Roth. 1998. Learning to resolve natural language ambiguities: A unified approach. In *proc. of the Fifteenth National Conference on Artificial Intelligence*, pages 806–813, Menlo Park, CA, USA, July. AAAI Press.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In 18^{th} *International Conference on Computational Linguistics (COLING)*, pages 947–953, July.