# Learning Methods for Combining Linguistic Indicators to Classify Verbs

**Eric V. Siegel**
Department of Computer Science
Columbia University
New York, NY 10027
evs@cs.columbia.edu

## Abstract

Fourteen linguistically-motivated numerical indicators are evaluated for their ability to categorize verbs as either *states* or *events*. The values for each indicator are computed automatically across a corpus of text. To improve classification performance, machine learning techniques are employed to combine multiple indicators. Three machine learning methods are compared for this task: decision tree induction, a genetic algorithm, and log-linear regression.

## 1 Introduction

The ability to distinguish states, e.g., *"Mark seems happy,"* from events, e.g., *"Renée ran down the street,"* is a necessary prerequisite for interpreting certain adverbial adjuncts, as well as identifying temporal constraints between sentences in a discourse (Moens and Steedman, 1988; Dorr, 1992; Klavans, 1994). Furthermore, *stativity* is the first of three fundamental temporal distinctions that compose the *aspectual class* of a clause. Aspectual classification is a necessary component for a system that analyzes temporal constraints, or performs lexical choice and tense selection in machine translation (Moens and Steedman, 1988; Passonneau, 1988; Dorr, 1992; Klavans, 1994).

Researchers have used empirical analysis of corpora to develop linguistically-based numerical indicators that aid in aspectual classification (Klavans and Chodorow, 1992; Siegel and McKeown, 1996). Specifically, this technique takes advantage of linguistic constraints that pertain to aspect, e.g., only clauses that describe an event can appear in the progressive. Therefore, a verb that appears more frequently in the progressive is more likely to describe an event.

In this paper, we evaluate fourteen quantitative linguistic indicators for their ability to classify verbs according to stativity. The values of these indicators are computed automatically across a corpus of text. Classification performance is then measured over an unrestricted set of verbs. Our analysis reveals a predictive value for several indicators that have not traditionally been linked to stativity in the linguistics literature. Then, in order to improve classification performance, we apply machine learning methods to combine multiple indicators. Three machine learning techniques are compared for this task: decision tree induction, a genetic algorithm, and log-linear regression.

In the following sections, we further detail and motivate the distinction between states and events. Next, we describe our approach, detailing the set of linguistic indicators, the corpus and tools used, and the machine learning methods. Finally, we present experimental results and discuss conclusions and future work.

## 2 Stative and Event Verbs

Stativity must be identified to detect temporal constraints between clauses attached with *when*. For example, in interpreting, *"She had good strength* **when** *objectively tested,"*[1] the *have*-**state** began before or at the beginning of the *test*-event, and ended after or at the end of the *test*-event. However, in interpreting, *"Phototherapy was discontinued* **when** *the bilirubin came down to 13,"* the *discontinue*-**event** began at the end of the *come*-event. As another example, the simple present reading of an event, e.g., *"He jogs,"* denotes the *habitual* reading, i.e., *"every day,"* whereas the simple present reading of a state, e.g., *"He appears healthy,"* implies *"at the moment."*

Identifying stativity is the first step toward aspectually classifying a clause. Events are further distinguished by two additional features: 1) telic events have an explicit culminating point in time, while non-telic events do not, and 2) extended events have a time duration, while atomic events do not. Detecting the telicity and atomicity of a clause is necessary to identify temporal constraints between clauses and to interpret certain adverbial adjuncts (Moens

[1] These examples of *when* come from the corpus of medical discharge summaries used for this work.

| If a verb can occur: | ...then it must be: |
|---|---|
| in the progressive | **Extended Event** |
| with a temporal adverb (e.g., *then*) | **Event** |
| with a duration *in*-PP (e.g., *in an hour*) | **Telic Event** |
| in the perfect tense | **Telic Event** or **State** |

Table 1: Example linguistic constraints excerpted from Klavans (1994).

and Steedman, 1988; Passonneau, 1988; Dorr, 1992; Klavans, 1994). However, since these features apply only to events and not to states, a clause first must be classified according to stativity.

Certain features of a clause, such as adjuncts and tense, are constrained by and contribute to the aspectual class of the clause (Vendler, 1967; Dowty, 1979; Pustejovsky, 1991; Passonneau, 1988; Klavans, 1994). Examples of such constraints are listed in Table 1. Each entry in this table describes a syntactic aspectual *marker* and the constraints on the aspectual class of any clause that appears with that marker. For example, a telic event can be modified by a duration *in*-PP, as in *"You found us there* **in ten minutes**,*"* but a state cannot, e.g., *"*You loved him* **in ten minutes**.*"*

In general, the presence of these linguistic markers in a particular clause indicates a constraint on the aspectual class of the clause, but the absence thereof does not place any constraint. This makes it difficult for a system to aspectually classify a clause based on the presence or absence of a marker. Therefore, these linguistic constraints are best exploited by a system that measures their frequencies across verbs.

Klavans and Chodorow (1992) pioneered the application of statistical corpus analysis to aspectual classification by placing verbs on a "stativity scale" according to the frequency with which they occur in the progressive. This way, verbs are automatically ranked according to their propensity towards stativity. We have previously applied this principle towards distinguishing telic events from non-telic events (Siegel and McKeown, 1996). Classification performance was increased by combining multiple aspectual markers with a genetic algorithm.

## 3 Approach

Our goal is to exploit linguistic constraints such as those listed in Table 1 by counting their frequencies in a corpus. For example, it is likely that event verbs will occur more frequently in the progressive than state verbs, since the progressive is constrained to occur with event verbs. Therefore, the frequency with which a verb occurs in the progressive *indicates* whether it is an event or stative verb.

We have evaluated 14 such linguistic indicators over clauses selected uniformly from a text corpus. In this way, we are measuring classification performance over an unrestricted set of verbs. First, the ability for each indicator to individually distinguish between stative and event verbs is evaluated. Then, in order in increase classification performance, machine learning techniques are employed to combine multiple indicators.

In this section, we first describe the set of linguistic indicators used to discriminate events and states. Then, we show how machine learning is used to combine multiple indicators to improve classification performance. Three learning methods are compared for this task. Finally, we describe the corpus and evaluation set used for these experiments.

### 3.1 Linguistic Indicators

The first column of Table 2 lists the 14 linguistic indicators evaluated in this paper for classifying verbs. The second and third columns show the average value for each indicator over stative and event verbs, respectively, as computed over a corpus of parsed clauses, described below in Section 3.3. These values, as well as the third column, are further detailed in Section 4.

Each verb has a unique value for each indicator. The first indicator, `frequency`, is simply the the frequency with which each verb occurs. As shown in Table 2, stative verbs occur more frequently than event verbs in our corpus.

The remaining 13 indicators measure how frequently each verb occurs in a clause with the linguistic marker indicated. This list includes the four markers listed in Table 1, as well as 9 additional markers that have not previously been linked to stativity. For example, the next three indicators listed in Table 2 measure the frequency with which verbs 1) are modified by *not* or *never*, 2) are modified by a *temporal* adverb such as *then* or *frequently*, and 3) have no deep subject (passivized phrases often have no deep subject, e.g., *"She was admitted to the hospital"*). As shown, stative verbs are modified by *not* or *never* more frequently than event verbs, but event verbs are modified by *temporal* adverbs more frequently than stative verbs. For further detail regarding the set of 14 indicators, see Siegel (1997).

An individual indicator can be used to classify verbs by simply establishing a threshold; if a verb's indicator value is below the threshold, it is assigned one class, otherwise it is assigned the alternative class. For example, in Table 3, which shows the predominant class and four indicator values corresponding to each of four verbs, a threshold of 1.00% would allow events to be distinguished from states based on the values of the *not/never* indicator. The next subsection describes how all 14 indicators can be used together to classify verbs.

| Verb | class | `freq` | *"not"* or *"never"* | *temporal* adverb | no deep subject |
|---|---|---|---|---|---|
| *show* | state | 2,131 | 1.55% | 0.52% | 18.07% |
| *admit* | event | 1,895 | 0.05% | 1.11% | 91.13% |
| *discharge* | event | 1,608 | 0.50% | 1.87% | 96.64% |
| *feel* | state | 1,177 | 4.61% | 1.20% | 52.52% |

Table 3: Example verbs and their indicator values.

| Linguistic Indicator | Stative Mean | Event Mean | T-test P-value |
|---|---|---|---|
| `frequency` | `932.89` | `667.57` | 0.0000 |
| *"not"* or *"never"* | 4.44% | 1.56% | 0.0000 |
| *temporal* adverb | 1.00% | 2.70% | 0.0000 |
| no deep subject | 36.05% | 57.56% | 0.0000 |
| past/pres participle | 20.98% | 15.37% | 0.0005 |
| duration *in*-PP | 0.16% | 0.60% | 0.0018 |
| perfect | 2.27% | 3.44% | 0.0054 |
| present tense | 11.19% | 8.94% | 0.0901 |
| progressive | 1.79% | 2.69% | 0.0903 |
| *manner* adverb | 0.00% | 0.03% | 0.1681 |
| *evaluation* adverb | 0.69% | 1.19% | 0.1766 |
| past tense | 62.85% | 65.69% | 0.2314 |
| duration *for*-PP | 0.59% | 0.61% | 0.8402 |
| *continuous* adverb | 0.04% | 0.03% | 0.8438 |

Table 2: Indicators discriminate between two classes.

### 3.2 Combining Indicators with Learning

Given a verb and its 14 indicator values, our goal is to use all 14 values in combination to classify the verb as a state or an event. Once a function for combining indicator values has been established, previously unobserved verbs can be automatically classified according to their indicator values. This section describes three machine learning methods employed to this end.

**Log-linear regression.** As suggested by Klavans and Chodorow (1992), a weighted sum of multiple indicators that results in one "overall" indicator may provide an increase in classification performance. This method embodies the intuition that each indicator correlates with the probability that a verb describes an event or state, but that each indicator has its own unique scale, and so must be weighted accordingly. One way to determine these weights is log-linear regression (Santner and Duffy, 1989), a popular technique for binary classification. This technique, which is more extensive than a simple weighted sum, applies an inverse logit function, and employs the iterative reweighted least squares algorithm (Baker and Nelder, 1989).

**Genetic programming.** An alternative to avoid the limitations of a linear combination is to generate a non-linear function tree that combines multiple indicators. A popular method for generating such function trees is a genetic algorithm (Holland, 1975; Goldberg, 1989). The use of genetic algorithms to generate function trees (Cramer, 1985; Koza, 1992) is frequently called genetic programming. The function trees are generated from a set of 17 primitives: the binary functions ADD, MULTIPLY and DIVIDE, and 14 terminals corresponding to the 14 indicators listed in Table 2. This set of primitives was established empirically; conditional functions, subtraction, and random constants failed to change performance significantly. The polarities for several indicators were reversed according to the polarities of the weights established by log-linear regression. Because the genetic algorithm is stochastic, each run may produce a different function tree. Runs of the genetic algorithm have a population size of 500, and end after 50,000 new individuals have been evaluated.

A threshold must be selected for both linear and function tree combinations of indicators. This way, overall outputs can be discriminated such that classification performance is maximized. For both methods, this threshold is established over the training set and frozen for evaluation over the test set.

**Decision trees.** Another method capable of modeling non-linear relationships between indicators is a decision tree. Each internal node of a decision tree is a choice point, dividing an individual indicator into ranges of possible values. Each leaf node is labeled with a classification (state or event). Given the set of indicator values corresponding to a verb, that verb's class is established by deterministically traversing the tree from the root to a leaf. The most popular method of decision tree induction, employed here, is recursive partitioning (Quinlan, 1986; Breiman et al., 1984), which expands the tree from top to bottom. The Splus statistical package was used for the induction process, with parameters set to their default values.

Previous efforts in corpus-based natural language processing have incorporated machine learning methods to coordinate multiple linguistic indicators, e.g., to classify adjectives according to markedness (Hatzivassiloglou and McKeown, 1995), to perform accent restoration (Yarowsky, 1994), for disambiguation problems (Yarowsky, 1994; Luk, 1995),

|  | n | States | Events |
|---|---|---|---|
| *be* | 23,409 | 100.0% | 0.0% |
| *have* | 7,882 | 69.9% | 30.1% |
| all other verbs | 66,682 | 16.2% | 83.8% |

Table 4: Breakdown of verb occurrences.

and for the automatic identification of semantically related groups of words (Pereira, Tishby, and Lee, 1993; Hatzivassiloglou and McKeown, 1993). For more detail on the machine learning experiments described here, see Siegel (1997).

### 3.3 A Parsed Corpus

The automatic identification of individual constituents within a clause is necessary to compute the values of the linguistic indicators in Table 2. The English Slot Grammar (ESG) (McCord, 1990) has previously been used on corpora to accumulate aspectual data (Klavans and Chodorow, 1992). ESG is particularly attractive for this task since its output describes a clause's deep roles, detecting, for example, the deep subject and object of a passivized phrase.

Our experiments are performed across a 1,159,891 word corpus of medical discharge summaries from which 97,973 clauses were parsed fully by ESG, with no self-diagnostic errors (ESG produced error messages on some of this corpus' complex sentences). The values of each indicator in Table 2 are computed, for each verb, across these 97,973 clauses.

In this paper, we evaluate our approach over verbs other than *be* and *have*, the two most frequent verbs in this corpus. Table 4 shows the distribution of clauses with *be*, *have*, and remaining verbs as their main verb. Clauses with *be* as their main verb always denote states. *Have* is highly ambiguous, so the aspectual classification of clauses headed by *have* must incorporate additional constituents. For example, *"The patient had Medicaid"* denotes a state, while, *"The patient had an enema"* denotes an event. In separate work, we have shown that the semantic category of the direct object of *have* informs classification according to stativity (Siegel, 1997). Since the remaining problem is to increase the classification accuracy over the 68.1% of clauses that have main verbs other than *be* and *have*, all results are measured only across that portion of the corpus. As shown in Table 4, 83.8% of clauses with verbs other than *be* and *have* are events.

A portion of the parsed clauses must be manually classified to provide supervised training data for the three learning methods mentioned above, and to provide a separate set of test data with which to evaluate the classification performance of our system. To this end, we manually marked 1,851 clauses selected uniformly from the set of parsed clauses not headed by *be* or *have*. As a linguistic test to mark according to stativity, each clause was tested for readability with *"What happened was..."*[2] Of these, 373 were rejected because of parsing problems (verb or direct object incorrectly identified). This left 1,478 parsed clauses, which were divided equally into 739 training and 739 testing cases.

Some verbs can denote both states and events, depending on other constituents of the clause. For example, *show* denotes a state in *"His lumbar puncture showed evidence of white cells,"* but denotes an event in *"He showed me the photographs."* However, in this corpus, most verbs other than *have* are highly dominated by one sense. Of the 739 clauses included in the training set, 235 verbs occurred. Only 11 of these verbs were observed as both states and events. Among these, there was a strong tendency towards one sense. For example, *show* appears primarily as a **state**. Only five verbs - *say*, *state*, *supplement*, *describe*, and *lie*, were not dominated by one class over 80% of the time. Further, each of these were observed less than 6 times a piece, which makes the estimation of sense dominance inaccurate.

The limited presence of verbal ambiguity in the test set does, however, place an upper bound of 97.4% on classification accuracy, since linguistic indicators are computed over the main verb only.

## 4 Results

Since we are evaluating our approach over verbs other than *be* and *have*, the test set is only 16.2% states, as shown in Table 4. Therefore, simply classifying every verb as an event achieves an accuracy of 83.8% over the 739 test cases, since 619 are events. However, this approach classifies all stative clauses incorrectly, achieving a *stative recall* of 0.0%. This method serves as a baseline for comparison since we are attempting to improve over an *uninformed* approach.[3]

### 4.1 Individual Indicators

The second and third columns of Table 2 show the average value for each indicator over stative and event clauses, as measured over the 739 training examples. As described above, these examples exclude *be* and *have*. For example, 4.44% of stative clauses are modified by either *not* or *never*, but only 1.56% of event clauses were modified by these adverbs. The fourth column shows the results of T-tests that compare the indicator values over stative verbs to those over event verbs. For example, there is less than a 0.05% chance that the difference between stative and event means for the first four indicators listed

[2] This test was suggested by Judith Klavans (personal communication).

[3] Similar baselines for comparison have been used for many classification problems (Duda and Hart, 1973), e.g., part-of-speech tagging (Church, 1988; Allen, 1995).

is due to chance. Overall, this shows that the differences in stative and event averages are statistically significant for the first seven indicators listed ($p < .01$).

This analysis has revealed correlations between verb class and five indicators that have not been linked to stativity in the linguistics literature. Of the top seven indicators shown to have positive correlations with stativity, three have been linguistically motivated, as shown in Table 1. The other four were not previously hypothesized to correlate with aspectual class: (1) verb `frequency`, (2) occurrences modified by *"not"* or "never", (3) occurrences with no deep subject, and (4) occurrences in the past or present participle. Furthermore, the last of these seven, occurrences in the perfect tense, was not previously hypothesized to correlate with stativity in particular.

However, a positive correlation between indicator value and verb class does not necessarily mean an indicator can be used to increase classification accuracy. Each indicator was tested individually for its ability to improve classification accuracy over the baseline by selecting the best classification threshold over the training data. Only two indicators, verb `frequency`, and occurrences with *not* and *never*, were able to improve classification accuracy over that obtained by classifying all clauses as events. To validate that this improved accuracy, the thresholds established over the training set were used over the test set, with resulting accuracies of 88.0% and 84.0%, respectively. Binomial tests showed the first of these to be a significant improvement over the baseline of 83.8%, but not the second.

### 4.2 Combining Indicators

All three machine learning methods successfully combined indicator values, improving classification accuracy over the baseline measure. As shown in Table 5, the decision tree's accuracy was 93.9%, genetic programming's function trees had an average accuracy of 91.2% over seven runs, and the log-linear regression achieved an 86.7% accuracy. Binomial tests showed that both the decision tree and genetic programming achieved a significant improvement over the 88.0% accuracy achieved by the `frequency` indicator alone. Therefore, we have shown that machine learning methods can successfully combine multiple numerical indicators to improve the accuracy by which verbs are classified.

The differences in accuracy between the three methods are each significant ($p < .01$). Therefore, these results highlight the importance of how linear and non-linear interactions between numerical linguistic indicators are modeled.

### 4.3 Improved Recall Tradeoff

The increase in the number of stative clauses correctly classified, i.e. stative recall, illustrates a more dramatic improvement over the baseline. As shown in Table 5, stative recalls of 74.2%, 47.4% and 34.2% were achieved by the three learning methods, as compared to the 0.0% stative recall achieved by the baseline, while only a small loss in recall over event clauses was suffered. The baseline does not classify any stative clauses correctly because it classifies all clauses as events. This difference in recall is more dramatic than the accuracy improvement because of the dominance of event clauses in the test set.

This favorable tradeoff between recall values presents an advantage for applications that weigh the identification of stative clauses more heavily than that of event clauses. For example, a prepositional phrase denoting a duration with *for*, e.g., *"for a minute,"* describes the duration of a state, e.g., *"She felt sick for two weeks,"* or the duration of the state that results from a telic event, e.g., *"She left the room for a minute."* That is, correctly identifying the use of *for* depends on identifying the stativity of the clause it modifies. A language understanding system that incorrectly classifies *"She felt sick for two weeks"* as a non-telic event will not detect that *"for two weeks"* describes the duration of the *feel*-state. If this system, for example, summarizes durations, it is important to correctly identify states. In this case, our approach is advantageous.

## 5 Conclusions and Future Work

We have compiled a set of fourteen quantitative linguistic indicators that, when used together, significantly improve the classification of verbs according to stativity. The values of these indicators are measured automatically across a corpus of text.

Each of three machine learning techniques successfully combined the indicators to improve classification performance. The best of the three, decision tree induction, achieved a classification accuracy of 93.9%, as compared to the uninformed baseline's accuracy of 83.8%. Furthermore, genetic programming and log-linear regression also achieved improvements over the baseline. These results were measured over an unrestricted set of verbs.

The improvement in classification performance is more dramatically illustrated by the favorable tradeoff between stative and event recall achieved by all three of these methods, which is profitable for tasks that weigh the identification of states more heavily than events.

This analysis has revealed correlations between stativity and five indicators that are not traditionally linked to stativity in the linguistic literature. Furthermore, one of these four, verb frequency, individually increased classification accuracy from the baseline method to 88.0%.

To classify a clause, the current system uses only the indicator values corresponding to the clause's main verb. This procedure could be expanded to

|  | overall | States |  | Events |  |
|---|---|---|---|---|---|
|  | accuracy | recall | precision | recall | precision |
| decision tree | 93.9% | 74.2% | 86.4% | 97.7% | 95.1% |
| genetic programming | 91.2% | 47.4% | 97.3% | 99.7% | 90.7% |
| log-linear | 86.7% | 34.2% | 68.3% | 96.9% | 88.4% |
| baseline | 83.8% | 0.0% | 100.0% | 100.0% | 83.8% |

Table 5: Comparison of three learning methods and a performance baseline.

incorporate rules that classify a clause directly from clausal features (e.g., Is the main verb *show*, is the clause in the progressive?), or by calculating indicator values over other clausal constituents in addition to the verb (Siegel and McKeown, 1996; Siegel, 1997).

Classification performance may also improve by incorporating additional linguistic indicators, such as co-occurrence with *rate* adverbs, e.g., *quickly*, or occurrences as a complement of *force* or *persuade*, as suggested by Klavans and Chodorow (1992).

## Acknowledgments

Kathleen R. McKeown was extremely helpful regarding the formulation of our work and Judith Klavans regarding linguistic techniques. Alexander D. Chaffee, Vasileios Hatzivassiloglou, Dragomir Radev and Dekai Wu provided many helpful insights regarding the evaluation and presentation of our results.

This research is supported in part by the Columbia University Center for Advanced Technology in High Performance Computing and Communications in Healthcare (funded by the New York State Science and Technology Foundation), the Office of Naval Research under contract N00014-95-1-0745 and by the National Science Foundation under contract GER-90-24069.

Finally, we would like to thank Andy Singleton for the use of his GPQuick software.

## References

Allen, J. 1995. *Natural Language Understanding.* Benjamin/Cummings, Redwood City, CA.

Baker, R.J. and J.A. Nelder. 1989. *The GLIM System, Release 3: Generalized Linear Interactive Modeling.* Numerical Algorithms Group, Oxford.

Breiman, L., J.H. Friedman, R.A. Olshen, and C.J. Stone. 1984. *Classification and Regression Trees.* Wadsworth, Belmont.

Church, K. 1988. A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the 2nd Conference for Applied Natural Language Processing*, pages 136–143.

Cramer, N. 1985. A representation for the adaptive generation of simple sequential programs. In J. Grefenstette, editor, *Proceedings of the [First] International Conference on Genetic Algorithms.* Lawrence Erlbaum.

Dorr, B.J. 1992. A two-level knowledge representation for machine translation: lexical semantics and tense/aspect. In James Pustejovsky and Sabine Bergler, editors, *Lexical Semantics and Knowledge Representation.* Springer Verlag, Berlin.

Dowty, D. 1979. *Word Meaning and Montague Grammar.* D. Reidel, Dordrecht, W. Germany.

Duda, R. O. and P.E. Hart. 1973. *Pattern Classification and Scene Analysis.* Wiley, New York.

Goldberg, D. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley Publishing Company, Inc., Reading, MA.

Hatzivassiloglou, V. and K.R. McKeown. 1993. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 172–182, Columbus, Ohio, June. Association for Computational Linguistics.

Hatzivassiloglou, V. and K.R. McKeown. 1995. A quantitative evaluation of linguistic tests for the automatic prediction of semantic markedness. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 197–204, Boston, Massachusetts, June. Association for Computational Linguistics.

Holland, J. 1975. *Adaptation in Natural and Artificial Systems.* The University of Michigan Press, Ann Arbor, MI.

Klavans, J.L. 1994. Linguistic tests over large corpora: aspectual classes in the lexicon. Technical report, Columbia University Dept. of Computer Science. unpublished manuscript.

Klavans, J.L. and M. Chodorow. 1992. Degrees of stativity: the lexical representation of verb aspect. In *Proceedings of the 14th International Conference on Computation Linguistics.*

Koza, J.R. 1992. *Genetic Programming: On the programming of computers by means of natural selection.* MIT Press, Cambridge, MA.

Luk, A.K. 1995. Statistical sense disambiguation with relatively small corpora using dictionary definitions. In *Proceedings of the 33rd Annual Meeting of the ACL*, Columbus, Ohio, June. Association for Computational Linguistics.

McCord, M.C. 1990. Slot grammar: A system for simpler construction of practical natural language grammars. In R. Studer, editor, *International Symposium on Natural Language and Logic*. Springer Verlag.

Moens, M. and M. Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2).

Passonneau, R.J. 1988. A computational model of the semantics of tense and aspect. *Computational Linguistics*, 14(2).

Pereira, F., N. Tishby, and L. Lee. 1993. Distributional clustering of English words. In *Proceedings of the 31st Annual Meeting of the ACL*, pages 183–190, Columbus, Ohio, June. Association for Computational Linguistics.

Pustejovsky, J. 1991. The syntax of event structure. *Cognition*, 41(103):47–92.

Quinlan, J.R. 1986. Induction of decision trees. *Machine Learning*, 1(1):81–106.

Santner, T.J. and D.E. Duffy. 1989. *The Statistical Analysis of Discrete Data*. Springer-Verlag, New York.

Siegel, E.V. 1997. *Classifying Natural Language Phrases with Corpus-Based Linguistic Indicators*. Ph.D. thesis, Columbia University.

Siegel, E.V. and K.R. McKeown. 1996. Gathering statistics to aspectually classify sentences with a genetic algorithm. In K. Oflazer and H. Somers, editors, *Proceedings of the Second International Conference on New Methods in Language Processing*, Ankara, Turkey, Sept. Bilkent University.

Vendler, Z. 1967. Verbs and times. In *Linguistics in Philosophy*. Cornell University Press, Ithaca, NY.

Yarowsky, D. 1994. Decision lists for lexical ambiguity resolution: Application to accent restoration in spanish and french. In *Proceedings of the 32nd Annual Meeting of the ACL*, San Francisco, CA, June. Morgan Kaufmann.