Conditional random fields (CRFs) are a probabilistic framework for labeling and segmenting structured data, such as sequences, trees and lattices. The underlying idea is that of defining a conditional probability distribution over label sequences given a particular observation sequence, rather than a joint distribution over both label and observation sequences. The primary advantage of CRFs over hidden Markov models is their conditional nature, resulting in the relaxation of the independence assumptions required by HMMs in order to ensure tractable inference. Additionally, CRFs avoid the label bias problem, a weakness exhibited by maximum entropy Markov models (MEMMs) and other conditional Markov models based on directed graphical models. CRFs outperform both MEMMs and HMMs on a number of real-world tasks in many fields, including bioinformatics, computational linguistics and speech recognition.

To determine whether a token is a name, our system uses weak evidence from a number of sources. The main consideration in deciding which information sources to use was the difficulty associated with creating and maintaining such sources. A secondary consideration was to keep the learned models as small as possible. The knowledge sources, encoded as a set of 29 features, are: Word Level Features: Language- or genre-specific cues can sometimes be exploited to provide evidence for name detection (e.g., in English, names are often capitalized). The following features are used in encoding tokens and the system learns which of these correlate strongly with names: (1) all-uppercase, (2) initial-caps, (3) all-numbers, (4) alphanumeric, (5) single-char, (6) single-s (if the token is the character “s”), and (7) single-i (if the token is the character “I”). Individually, none of the local word-level features are very effective: the strongest individual feature is all-caps; it flags 474 tokens in the training set (containing 50,416 tokens, of which 3,632 are names). Of these, 449 are actually names, the rest being non-name acronyms such as “CEO” and “PC”, yielding an F1-score2 of only 21 (Precision = 94; Recall = 12). The feature initial-caps is similar, flagging 5,650 words, of which 3,572 are names, leading to an F1-score of 82 (Precision = 73; Recall = 94). Note that not all capitalized words are names, and that not all names are capitalized (e.g., “van Gogh”). For English text, capitalization, in conjunction with reliable end-of-sentence boundary detection, is a good indicator for names. However, determining sentence boundaries is difficult since common boundaries such as periods, question- and exclamation-marks can occur in many different contexts [16; 13]. While the system does not explicitly contain rules for sentence boundary analysis, by using contextual cues, it can account for many sentence boundaries. Dictionary Look-Up: Another weak heuristic for determining whether a particular token is a name is to check whether it can be found in a dictionary. Since many names are not valid English words, this resource can identify some potential names. The dictionary used in our experiments was the standard spelling dictionary available on most UNIX systems; it contained 45,402 words, of which 6,784 were capitalized, and were discarded as names. The remaining 38,618 tokens contained multiple morphological variants of the same word (further decreasing the number of unique root forms). Finally, since a number of English names are also part of the regular vocabulary (e.g., “mark”, “baker” and “stone”), name detection using only evidence from the dictionary is not very reliable: the F1-score for the dictionary module alone on our training set was only 64. Part-of-Speech Tagger: Part-of-Speech (POS) tags can be used by other modules to reason about the roles and relative importance of words/tokens in various contexts. In this system, we used the Brill tagger for POS tagging3 . Brill reports approximately 97% overall accuracy for words in the WSJ corpus for the tagger [6; 7]. Its performance is lower on the named-entity task: on our training data, the tagger obtained an F1-score of 83 (P = 81, R = 86); consistent results are reported in [1]. The following POS tags were used as features by the machine learning component of our system: (1) determiner, (2) foreign-word, (if the token is one that the tagger has not seen), (3) preposition, (4) adjective, (5) noun, (6) proper-noun, (7) personal-pronoun, (8) possessive-pronoun, (9) verb, (10) WH-pronoun (which, what, etc.), (11) unknown-POS. Punctuation: Robust name detection probably requires that the system capture contextual syntactic information. (At the very least, to disambiguate capitalization cues due to sentence boundaries.) The system learns syntactic patterns that may indicate named entities. Section 5 discusses the effects of varying the size of this contextual window. The following punctuation characters are encoded as features: (1) comma; (2) period; (3) exclamation mark; (4) question mark; (5) semi-colon; (6) colon; (7) plus or minus sign; (8) apostrophe; (9) left parenthesis; (10) right parenthesis.