1 Introduction

Fluency is the goal of most language learners. While mastering the spoken language is usually the primary goals for learners, becoming a proficient writer is of importance for many learners, particularly those whose learn languages for professional or academic reasons. Although it is generally less obvious than in the case of spoken language, most learners will reach a stage where they produce grammatically-correct language, but which is still identifiable as nonnative. At this stage, further improvement of their language is no longer a matter of error correction, but of making subtle changes which may not be apparent to the learner. It would be useful if there existed a learning tool that could analyze written language and give feedback to the user on which aspects of their language need to be modified to reach a native level. Such a tool would only be possible if there existed an automated system to could analyze language, classify as native or nonnative, and give the reasons for the classification. This paper describes such a system.

In the past, most computer-based language learner tools have been for beginning and intermediate learners, focusing on grammatical mistakes. Such tools analyze text and offer suggestions on things such as correct article usage or noun pluralization. In many ways these tools are very similar to the grammar checkers found in most modern word-processing packages, though those tend to be targeted towards native speaker, or towards no particular group of users. In general, these tools are not of any use on syntactically well-formed texts.

The system explored in this study relies on modern language parsing systems. Probabilistic parsers are capable of generating grammatical parse trees of texts in a number of languages, particularly English, with a high degree of accuracy. Such systems, in particular the Stanford Parser, used in this study, are often capable of presenting additional syntactic information in the form of dependency graphs, which indicate various relationships between words.

This study uses automatically generated parse trees and dependency graphs to identify grammatical features. The relative frequency of each feature is calculated, and this infor-

mation is used to train automatic classifiers, which are then tested on other text samples.

This paper describes the methods in which grammatical features are extracted from the output of the Stanford Parser and analyzes the effectiveness of classifiers trained on these features. Three separate experiments are performed, each considering a different set of features. Two of these experiments further break down feature sets into smaller related subsets to ease in the analysis of their effectiveness. The decision trees generated by the classifiers are then explored to identify the linguistic significance of the features used in the trees. Finally, this paper proposes a design for a learner's tool based on this system and suggests other possible applications as well.