Designing an Intelligent Discourse Evaluation Tool: Theoretical, Empirical, and Technological Considerations

Elena Cotos
Department of English
Iowa State University

Considering the promising potential of ICALL and intelligent feedback, this paper addresses this topic from the perspective of practical applications. It reviews existing intelligent language learning systems and their feedback capabilities, arguing that while ICALL is very promising and highly complex, its development should be more principled. In other words, the decisions behind the system design need to be justified theoretically, integrating technology with theory, research, and practice in instructed SLA. To support this argument, this paper provides an example of how this was achieved in the context of L2 graduate academic writing. It describes a new natural language processing-based application, IADE (Intelligent Academic Discourse Evaluator), explaining how pedagogical and theoretical perspectives were synthesized, moving from practice to theory and from theory to design. Empirical findings on the provision of feedback and project-specific discourse analysis research also informed the development of IADE so that it meets the particular needs of the targeted learners. From the technological point of view, the program’s overall conceptualization and operationalization draw from the principles of Evidence Centered Design. Since development generally leads to implementation, the paper concludes with issues related to ICALL evaluation.

Second language acquisition (SLA) research, in consensus with views in psychology, has long argued that individuals learn languages differently, depending on such influencing factors as idiosyncratic learning strategies, cognitive styles, and various affective factors (Dörnyei & Skehan, 2003). It has been suggested that it is not in the power of a human teacher to adapt to all learner differences and to provide individualized instruction to groups of students; therefore, computers have been proposed as a powerful and practical alternative (Britt, 1967; Suppes, 1967).

In the past decades, Computer Assisted Language Learning (CALL) has witnessed an increasing interest due to its ability to provide individualized instruction through immediate feedback. Garret (1987) identifies four types of CALL feedback: (1) only the correct answer is presented; (2) the location of errors on the basis of a letter-by-letter comparison of the learner’s input with the machine-stored correct version is indicated; (3) based on an analysis of the anticipated incorrect answers, error messages associated with possible errors are stored in the computer and are presented if the learner’s response matches those possible errors; and (4) based on a linguistic analysis of the learner’s response compared to an analysis derived from relevant grammar rules and lexicon of the target language, problematic or missing items are pinpointed. The last type of feedback, known as intelligent (or ICALL) feedback, is the most sophisticated in that it uses the Natural Language Processing (NLP) approach, which is superior to the so-called ‘pattern
markup’ and ‘error-anticipation’ techniques used in other conventional types of feedback because it is capable of diagnosing errors, providing detailed explanations about the nature of those errors, and responding simultaneously to more than one problematic aspect of language use that may occur in learner’s production.

NLP techniques are core to a number of ICALL programs, e.g., CriterionSM by Educational Testing Service, WriteToLearn by Pearson, and MyAccess! by Vantage Learning, all of which aim at providing means for error remediation and at enhancing learning opportunities by allowing students to submit their text drafts, receive feedback, act upon it, improve, and re-submit their texts iteratively in expectation of better outcomes. Considering the promising potential of intelligent feedback, this paper explores this topic from the perspective of practical applications. First, a brief overview of existing NLP-based language learning systems is provided. It is also argued that while ICALL is highly complex, it is generally not theoretically informed. Accounting for this limitation, the next section introduces a new web-based ICALL application, named IADE (Intelligent Academic Discourse Evaluator) and describes how pedagogical theoretical perspectives were synthesized, moving from practice to theory and from theory to design. The empirical knowledge that informed the decisions regarding IADE’s feedback as well as the research steps taken to build this program are also covered here. Finally, recommendations for informed ICALL design are offered, along with a call for further research on intelligent systems’ effectiveness.

AN OVERVIEW OF ICALL SYSTEMS

Developing ICALL systems is not a new endeavor. The beginning of the 1990s was very productive in this respect, a number of NLP-based applications being piloted in that period of time. For example, Grammar-Debugger (Chen & Luomai, 1990) was a parser designed to analyze grammatical as well as ungrammatical sentences produced by Chinese learners of English. Another parser, Syncheck (Juozulynas, 1994; Sanders, 1991) analyzed syntax to help learners write German compositions. GPARS (Loritz, 1992) included parsers for several languages such as English, Russian, Japanese, and Chinese. LINGER (Yazdani, 1991) relied on a combination of tools, e.g., a language-independent parsing mechanism, a language-specific dictionary, and a language-specific grammar, which allowed the system to cope with various aspects of Romance and Germanic languages. Levin, Evans, and Gates (1991) developed a multi-media workbench for learners of foreign languages (ALICE), which offered a set of tools for designing different types of intelligent programs in any language. ALICE’s facilities included tools and resources for syntactic parsing, morphological analysis, error detection, on-line dictionaries and other reference materials, as well as for indexing and retrieval of examples from video, audio, and text corpora. ALICE supported applications targeted at grammar and vocabulary drills, games and simulations, reading and writing assistants, and open-ended learning environments, its two prototypes being a Spanish writing assistant and a Japanese grammar exercise program.
Table 1. ICALL Systems

<table>
<thead>
<tr>
<th>ICALL system</th>
<th>Language</th>
<th>System components</th>
<th>Type of feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labrie &amp; Singh (1991) - Miniprof</td>
<td>French</td>
<td>parser, error diagnosis, tutor</td>
<td>dialog, error-specific, explicit pointer to error, metalinguistic</td>
</tr>
<tr>
<td>Liou (1991)</td>
<td>English</td>
<td>pattern matching, parser, message generator</td>
<td>error-specific, explicit pointer to error, metalinguistic examples</td>
</tr>
<tr>
<td>Holland et al. (1993) - BRIDGE</td>
<td>German</td>
<td>parser, graphical aids</td>
<td>right/wrong, error-specific, explicit pointer to error, metalinguistic examples</td>
</tr>
<tr>
<td>Nagata (1993) - NIHONGO-CALI</td>
<td>Japanese</td>
<td>morphological parser, syntactic parser, core lexicon, morphological rules, grammar rules</td>
<td>error-specific, explicit pointer to error, metalinguistic examples</td>
</tr>
<tr>
<td>Yang &amp; Akahori (1998)</td>
<td>Japanese</td>
<td>WWW interface, morpheme analyzer, syntax analyzer, analyzer; grammar rules, feedback system, knowledge database, students’ learning histories, dictionary</td>
<td>error-specific, explicit pointer to error, metalinguistic, examples, source links, adaptive individualized</td>
</tr>
<tr>
<td>Heift &amp; Nicholson, (2001) - German Tutor</td>
<td>English</td>
<td>analysis module, filtering module, Student Model</td>
<td>error-specific, explicit pointer to error, metalinguistic, correction</td>
</tr>
<tr>
<td>Toole &amp; Heift (2002) - ESL Tutor</td>
<td>Greek</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heift et al. (2000) - Greek Tutor</td>
<td>Greek</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chen &amp; Tokuda (2003), Tokuda &amp; Chen (2001, 2004) – Azalea</td>
<td>English</td>
<td>template automaton structure, diagnostic engine, POS parser, parser-based learner model, visual interface</td>
<td>error-specific, explicit pointer to error, metalinguistic, correction</td>
</tr>
<tr>
<td>Reuer (2003)</td>
<td>English</td>
<td>LFG parser, unrestricted grammar and lexicon, linguistic knowledge</td>
<td>error-specific, explicit pointer to error, metalinguistic, tree-form</td>
</tr>
<tr>
<td>Shaalan (2005) - Arabic ICALL</td>
<td>Arabic</td>
<td>user interface, course material, morphological analyzer, syntax parser, non/grammatical rules, lexicon, feedback module</td>
<td>error-specific, explicit pointer to error, metalinguistic</td>
</tr>
</tbody>
</table>

More recent implementations that employ intelligent feedback (see Table 1) are generally described from the computational and functional point of view, their description being limited to the presentation of system components and the function they perform. Some applications seem to have possible theoretical bases for their design, but those bases are never made explicit and may only be inferred.

Overall, what the ICALL programs briefly overviewed in this section have in common is that their design revolves around the functionality of a parser. Their feedback features,
generated by different system components and varying in their degree of specificity, explicitness and ability to adapt to individual learners, may have great instructional potential. The question, however, is not how to add sophistication to an ICALL system, but rather how to justify that sophistication. That is, the decisions behind the system design need to be justified theoretically (Levy, 1997), integrating technology with theory, research, and practice in instructed SLA (Chapelle, 2007a). This connection has been largely overlooked, but it is an issue of indisputable importance, and it will be further addressed with respect to the design and implementation of IADE.

THE INTELLIGENT ACADEMIC DISCOURSE EVALUATOR

The Intelligent Academic Discourse Evaluator (IADE) is an online NLP-based program that automatically analyzes learner drafts of research article introductions in terms of discourse development features. Given a draft, IADE evaluates it and returns immediate discipline-specific individualized feedback, engaging learners in an iterative revision process and, at the same time, enhancing the formative assessment aspect of the instructional process.

Targeted Learners

IADE was developed for learners who are non-native speaker graduate students at Iowa State University (ISU). They specialize in a wide range of disciplines, e.g., Biomedical Sciences, Chemical Engineering, Organic/Inorganic Chemistry, Statistics, Agronomy, Mathematics, Public Administration, Sociology, Curriculum and Instruction, to name a few. The students have various language backgrounds, such as Chinese, Hindi, Bengali, Spanish, Telugu, Vietnamese, Polish, Tamil, etc. The period of their stay in the United States ranges from a few weeks to several years. The students are enrolled in Master's and Doctoral programs, and, as international students, they are required to have achieved a minimum TOEFL score before they are admitted for graduate study. This score can be on a paper-based, computer-based, or internet-based test, and it varies depending on the department. The TOEFL scores required in graduate programs at ISU generally range from 530 to 640 on a paper-based test.

Upon admission, even if the TOEFL score is satisfactory, international graduate students are required to take the institutional English Placement Test (EPT) administered at the beginning of their first semester of enrollment. If they pass this test, they are exempt from English requirements at ISU. If they do not pass the EPT, they are placed in English courses. The EPT has a writing component, the task for which is to elaborate on a piece of data obtained through research. Students who, according to the scoring rubric, perform relatively well on this task, that is, express themselves with a certain degree of ease, effectively organize their thoughts, operate with varied vocabulary, and make use of complex grammatical structures, but have apparent weaknesses in developing academic discourse, are placed in the highest level academic writing course for international students.
**Instructional Context**

Academic writing courses for graduate NNS students should both provide them with general writing practice and “help initiate writers into their field-specific research communities” (Frodesen, 1995, p.33). In support of this opinion, Kushner (1997) also argues that graduate ESL courses have to combine language and discourse with the skill of writing within professional norms. Typically, universities in English-speaking countries offer academic writing courses that bring together students from various graduate programs, and such heterogeneity makes instruction very challenging. It would be ideal to have field-specific writing classes (Cargill, Cadman, & McGowan, 2001; Hyland, 2002), which would be designed for graduate students in certain disciplines, or to have instructors who would be members of students’ discourse communities and who would also be trained in language matters (Levis & Muller-Levis, 2003). In reality, however, issues of practicality interfere with the implementation of such solutions.

At ISU, the NNS graduate-level academic writing course has undergone different changes to bridge the gap between language instruction and disciplinary writing norms. Initially, it was a course that included a wide range of assignments, such as memos, letters, curriculum vitae, description of processes or mechanisms, literature reviews, research reports, critiques, and oral presentations. The number of assignments has gradually decreased, the major focus being placed on the research report genre. Vann and Myers (2001) described the inductive analysis of research writing in specific disciplines, in which students examine the format, content, and grammatical and rhetorical conventions of each section of the research report. Supplements to this approach were tasks that required students to write journal entries about the rhetorical conventions of prominent journals in their disciplines and tasks that placed the writing-up research experience “in the framework of an interactive, cooperative effort with cross-cultural interaction” (Vann & Myers, 2001, p. 82).

Later, after having followed a primarily skill-based approach, in which students wrote field-specific literature reviews, summaries, paraphrasing, data commentaries, and other discipline-specific texts, Levis and Muller-Levis (2003) reported on transforming the course into one focused on project-based writing. The project consisted of carrying out original research, the topic of which, for the purpose of coping with discipline diversity, was the same for all students and was determined by the instructor. From the start, the students were provided with a limited set of articles on, for instance, cross-cultural adjustment, with which they worked to identify potential research questions for further investigation and to write the literature review. This approach placed a heavy emphasis on collaboration as students worked in small groups on developing data-collection instruments and on data analysis. Oral presentations on the group research project wrapped up the course.

At present, the academic writing course in question is corpus-based and genre-based, combining a top-down approach to genre analysis and a bottom-up approach to the analysis of corpora (Cortes, 2007). Cortes (2007) explains that the course was redesigned
to better address the issues of genre-specificity and disciplinarity. In this version of the course, each student is provided with a corpus of research articles published in top journals of his/her discipline. Students conduct class analyses of their corpus according to guidelines from empirical findings in applied linguistics about the discourse tendencies in research article writing exhibited through ‘moves’, which are “discoursal or rhetorical units that perform a coherent communicative function in a written or spoken discourse” (Swales, 2004, p. 228). Their classwork is to discover organizational and linguistic patterns characteristic of their particular discipline, report on their observations, and apply the knowledge they gain from the corpus analyses when writing a research article for the final project of the course.

Practical and Theoretical Background for the Design of IADE

The task to write an academic paper is a major part of NNS graduate writing courses, irrespective of the teaching approach. It is also the most problematic, in that it requires sufficient practice on the part of the student and individualized, discipline-specific remediation on the part of the instructor. The fact that the approaches described above all fail to provide students with these options motivated the development of IADE. However, while practice was the main driving factor, it was complemented by a number of other important perspectives. Considerations as to what theoretical insights can be most informative vis-à-vis the targeted learners’ needs were central to the decisions regarding IADE’s design.

First and foremost, the targeted learners need to develop their writing skills. Focusing on skills, the fundamental claim of the Skill Acquisition Theory, a view on learning from psychology, (DeKeyser, 2007; Newell & Rosenbloom, 1981), is that “learning a wide variety of skills shows remarkable similarity in development from initial representation of knowledge through initial changes in behavior to eventual fluent, spontaneous, largely effortless, and highly skilled behavior” (DeKeyser, 2007, p. 97). Byrne (1986) postulated three stages of skill development: presentation, practice, and production. While the learners in graduate-level academic writing contexts are exposed to sufficient presentation in the classroom (meant to equip them with new knowledge), they lack opportunities for practice and production. Thus, the theoretical stipulation that repeatedly acting on the new knowledge is essential for gradual automatization of that knowledge dictates the use of an iterative process of revision with the help of the computer program.

Second, a closely related need of the targeted learners is to learn how to write academically, abiding by the writing conventions of their fields. Discourse analysis research has demonstrated that the development of the content and structure of research articles has its own specifics depending on the academic domain. Indeed, the genre of research articles is very complex. A particular degree of complexity is exhibited in Introduction sections, where the writer has to operate with various discourse techniques to fulfill a multitude of purposes. Swales’ (1981, 1990, 2004) seminal work on introduction sections offers exceptional guidance, specifying all possible discourse elements, called moves. Therefore, the move schema developed by Swales served as a model for the overall approach to the text analysis performed by IADE (see Appendix A).
Third, to achieve certain rhetorical purposes in their writing, learners need to know how to express functional meaning. Systemic Functional Linguistics (Halliday, 1985a, 1985b; Martin, 1992) views language as a system which is used to express particular meanings in particular contexts. In the case of writing research article introductions, particular aspects of the context define the functional meanings of moves as well as the language used to express those meanings. Therefore, as indicated by this language theory, IADE focuses on texts and their attendant contextual richness, rather than on sentences or other aspects of text. IADE also intends to establish relations between academic texts and the professional practice they realize and sustains the analysis of learner production.

Figure 1. Practical and theoretical background for IADE design

Last but not least, since the targeted learners are non-native speakers of English and since writing is a language skill, it is important to take into consideration what facilitates language learning. According to the interactionist theory of SLA, input, interaction, and output all play significant roles in second language acquisition (Gass, 1997; Long, 1996; Pica, 1994; Swain, 1993). Ellis (1994) expanded the idea of interaction from interpersonal interaction to that of intrapersonal interaction, i.e. “interaction that can occur in our minds, […] and, more covertly, when different modules of the mind interact to construct an understanding of or a response to some phenomena” (p.1). Chapelle (1998; 2007a) connected these concepts to learner-computer interaction, which, given that in this case the instruction is held in a computer-mediated environment, makes the interactionist theory of SLA relevant to, and supportive of IADE. Thus, the program analyzes learners’ output and transfers it into feedback input. At the same time, it aims at
encouraging noticing of and focus on discourse form by providing input enhancement of rhetorical moves in its color-coded feedback (see Figure 2). The feedback, in turn, is meant to foster intrapersonal interaction, which is expected to lead to the creation of new output. Figure 1 shows the connection between the above enumerated practical needs, theoretical perspectives, and implemented design decisions.

**Empirical Background for the Design of IADE**

*Previous research on feedback.* Considering the pedagogical intent for the implementation of IADE, decisions regarding its feedback were informed by research so that the program’s feedback is meaningful, i.e., a “response that provides a learning opportunity for students” (Heift, 2003, p. 533). Studies on feedback speak in favor of various types of feedback, e.g., explicit (Caroll & Swain, 1993; Lyster, 1998), individual specific (Hyland, 1998), metalinguistic (Rosa & Leow, 2004), negative cognitive (Long, 1996; Mitchell & Myles, 1998), detailed iterative (Hyland & Hyland, 2006), and accurate, short, one at a time (Van der Linden, 1993).

Intelligent feedback, i.e., NLP-generated explanations of the learners’ errors, was found to be more effective than traditional CALL feedback, i.e., messages only pointing to missing or unexpected words in the learners’ responses (Nagata, 1993). It appeared to be more effective even than the enhanced version of traditional feedback, which also indicates the positions of the missing items (Nagata, 1995). Similarly, Yang and Akahori’s (1999) findings suggest that feedback which corresponds to the input created by the learner is superior to feedback displaying the correct answer in a ‘multiple selection’ method in that it enhances self-correction. In terms of correction, van der Linden (1993), while examining the strategies learners employed when interacting with different levels of feedback, observed that learners felt motivated to self-correct when they received feedback about the type of error committed.

Furthermore, the more detailed the feedback is, the better the learning outcomes are (Heift, 2001, 2002, 2004, 2005). Therefore, the feedback should not only display the error, but also explain why the response is inappropriate. Recently, Heift (2008) explored the effects of two types of feedback, metalinguistic explanation and metalinguistic clue, on beginning, advanced-beginning, and low-intermediate learners of German, looking into the longitudinal effects of these types of feedback and into their impact on different error classes. Overall, she concluded that learners show significantly more learner uptake over time with the more error-specific/explicit feedback type. In another study (Heift & Rimrott, 2008), the researchers considered three distinct feedback types for spelling errors: metalinguistic with emphasis (feedback includes suggestions for the misspelled word in addition to displaying the incorrect sentence and highlighting the error), metalinguistic (feedback indicates that a spelling error has occurred and provides a list of suggestions), and repetition (feedback does not contain any suggestions for correction). Here, too, learners had the most correct responses and the greatest uptake with the most explicit and prominent feedback type, i.e., metalinguistic with emphasis, repetition being the least effective.
In sum, intelligent feedback is argued to be effective if it is: (a) individualized according to specific learner input, (b) pointing to the error type, (c) explicitly explaining the error, and (d) leading to self-correction. With this in mind, and considering the research findings regarding traditional feedback, IADE’s feedback possesses the following characteristics viewed as beneficial for language learning:

- **Immediate** – the feedback is provided immediately, in up to 60 seconds from the time of submission of the draft
- **Intelligent** – the feedback is generated automatically by a natural language processing-based engine
- **Individual-specific** – the feedback is provided to the student individually based on his/her input and on its comparison to the respective discipline
- **Metalinguistic** – the feedback is provided in definitional terms (i.e., “… of your sentences belong to Move 1”)
- **Short** – the feedback is concise in that it briefly presents the descriptive percentages representing the distribution of Moves in the students’ draft and in the introductions of his/her discipline (e.g., “This is below average compared to Move 1 in your discipline, where the minimum is 45.455%, the average is 65.799%, and the maximum is 87.097%)
- **Negative** – the feedback points to drawbacks in the discourse development of the draft (e.g., the feedback may say: “This is way below average (or below average/way above average/above average) compared to Move 1 in your discipline”)
- **Explicit** – the feedback is clearly stated through comments (“way above average”, “above average”, “average”, “below average”, “way below average”) and demonstrated through colors and percentages
- **Output-focused** – the feedback targets the student’s actual written production
- **Iterative** – the feedback is provided as often as requested
- **Color-coded** – for input enhancement, the feedback about the draft structure is provided in colors: blue for Move 1, orange for Move 2, and green for Move 3

A screenshot of the feedback provided by IADE is shown in Figure 2. Note how the colors represent the distribution of the moves (blue – move 1, orange – move 2, green – move 3). Numbers indicate how far from or how close to the norms in his/her discipline the student’s writing is.
Project-specific research on discourse moves. Equally important to the development of IADE is the empirically-obtained data that lies at the basis of the program’s automated evaluation and feedback. It was obtained through a thorough discourse analysis of a corpus of 1000 published research article introductions representative of 50 disciplines (see Appendix C), which was compiled from an existing corpus of published research articles used by students in the academic writing course described above. This corpus of introductions was manually annotated for moves and steps based on Swales’ framework presented in Appendix A.

Table 2. Inter-annotator agreement

<table>
<thead>
<tr>
<th></th>
<th>Move 1</th>
<th>Move 2</th>
<th>Move 3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Nr. agreed</strong></td>
<td>457</td>
<td>452</td>
<td>480</td>
</tr>
<tr>
<td><strong>P(A)</strong></td>
<td>0.938</td>
<td>0.928</td>
<td>0.986</td>
</tr>
<tr>
<td><strong>k</strong></td>
<td>0.931</td>
<td>0.919</td>
<td>0.984</td>
</tr>
</tbody>
</table>

**Nr. agreed** – number of sentences on which both annotators agreed  

**P(A)** – observed probability of agreement  

**k** – inter-annotator agreement

Annotation was performed at the sentence level, each sentence being assigned at least one move and a step within that move as specified in the markup scheme. It is worth mentioning that annotations were done after a second annotator marked a sample of files...
across all the disciplines adding up to 487 sentences and after inter-annotator agreement was calculated. Table 2 indicates relatively high agreements between the two annotators. Agreement between a human rater and IADE remains to be calculated.

<table>
<thead>
<tr>
<th>Steps in Computer Engineering research articles introductions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Move 1</strong></td>
</tr>
<tr>
<td>Claiming centrality: [min 0%, avg 6.21%, max 21.42%]</td>
</tr>
<tr>
<td>Making topic generalization/s: [min 0%, avg 26.71%, max 48.27%]</td>
</tr>
<tr>
<td>Reviewing previous research: [min 0%, avg 8.05%, max 36.36%]</td>
</tr>
<tr>
<td><strong>Move 2</strong></td>
</tr>
<tr>
<td>Indicating a gap: [min 0%, avg 0.91%, max 7.69%]</td>
</tr>
<tr>
<td>Highlighting a problem: [min 0%, avg 9.43%, max 36.84%]</td>
</tr>
<tr>
<td>Question-raising: [min 0%, avg 0%, max 0%]</td>
</tr>
<tr>
<td>Hypothesizing: [min 0%, avg 1.60%, max 13.33%]</td>
</tr>
<tr>
<td>Presenting justification: [min 0%, avg 2.56%, max 18.91%]</td>
</tr>
<tr>
<td><strong>Move 3</strong></td>
</tr>
<tr>
<td>Announcing present research descriptively: [min 0%, avg 16.65%, max 64.28%]</td>
</tr>
<tr>
<td>Announcing present research purposefully: [min 0%, avg 1.28%, max 7.14%]</td>
</tr>
<tr>
<td>Presenting research questions: [min 0%, avg 0%, max 0%]</td>
</tr>
<tr>
<td>Presenting hypotheses: [min 0%, avg 0%, max 0%]</td>
</tr>
<tr>
<td>Definitional clarifications: [min 0%, avg 0.56%, max 7.84%]</td>
</tr>
<tr>
<td>Summarizing methods: [min 0%, avg 4.54%, max 20%]</td>
</tr>
<tr>
<td>Announcing principal outcomes: [min 0%, avg 3.72%, max 13.33%]</td>
</tr>
<tr>
<td>Stating the value of the present research: [min 0%, avg 6.34%, max 27.77%]</td>
</tr>
<tr>
<td>Outlining the structure of the paper: [min 0%, avg 11.03%, max 33.33%]</td>
</tr>
</tbody>
</table>

Figure 3. Sample Step Statistics Provided Through IADE’s “Help Options”

The annotation scheme allowed for multiple layers of annotation for cases when the same sentence signified more than one move or more than one step. This made it possible to capture an array of the semantic shades rendered by a given sentence, which can be seen in the examples below:

<intro_m1 “In summarizing the research in this area, Fama and French’s (1998) assessment is that no consensus has yet emerged on the question of whether a dividend tax penalty is capitalized into the return on a firm’s common stock.” step= “reviewing previous research”>

<intro_m2 “no consensus has yet emerged on the question of whether a dividend tax penalty is capitalized into the return on a firm’s common stock.” step= “indicating a gap”>

(from the Accounting subcorpus [ACCN001])
<intro_m3 “Applying the experiment to several peptides” step=“summarizing methods”>

<intro_m3 “Applying the experiment to several peptides, we find that the experimental CSA principal clues agree with the calculated values to an average of 2.4 ppm, while the difference in the tensor orientation is less than 14.” step=“announcing principal outcomes”>

(from the Biophysics subcorpus [BIOP017])

This discourse analysis and annotation was a prerequisite for training IADE, the process of which is described in Pendar and Cotos (2008). It is also the source of the color-coded feedback generated by the program. The numerical feedback content draws from statistical analyses of move lengths and frequencies in each annotated discipline subcorpus. Similar statistics were calculated for the purpose of providing the students with a more detailed description of the discourse development of RA introductions at the level of steps. Specifically, aiming at stimulating learner-computer interaction and input modification, IADE offers learners “Help Options,” among which is the annotated corpus itself as well as statistical information about each step’s representation in their field. Figure 3 shows an example of such information for a student in Computer Engineering.

**Technological Background for the Design of IADE**

The development of IADE was guided by the principles of Evidence Centered Design (ECD), “an approach to constructing and implementing educational assessments in terms of evidentiary arguments” (Mislevy, Steinberg, Almond, & Lukas, 2006, p. 15). The ECD relies on Conceptual Assessment Framework (CAF) models for design and on a generic framework for assessment delivery. According to Mislevy, Almond, and Lukas (2003), the design models serve as a blueprint for the operational elements of the assessment. These models are very complicated and demanding from the technological point of view; however, the idea contained in the CAF Evidence model, due to its focus on observable evidence that describes learners’ performance as reflected in their work products, proved to be applicable in the case of IADE. Further, the same authors explain that the ECD delivery framework consists of four processes: (1) the Presentation Process, which is accountable for presenting the task and the supporting materials and for collecting learners’ work products; (2) the Response Processing, which is responsible for identifying the observable evidence in the work products; (3) the Summary Scoring, which should gather the outcomes across multiple tasks to eventually assign a score; and (4) the Activity Selection, the function of which is to make decisions regarding the next task or the time of final scoring. Considering that IADE provides one major task (to revise) and that it was meant for formative evaluation, only the first two processes seemed pertinent and were thus employed. Within this framework, an indispensable component is the Evidence Composite Library – a database where all the data that makes those processes work are contained. This component has also become integral in IADE.
The ECD influence is further reflected in the system overview (Figure 4) and description. In short, guided by ECD, the program was built to identify the discourse elements of students’ work products that constitute evidence and to characterize the strength of this evidence about targeted writing proficiencies for the purpose of formative assessment.

The Presentation Process module is operationalized through a web interface where users can access a sign-up page, a log-in page, and a drafting page with instructions, a resubmit button, and sign-out option. After the first submission, the drafting page displays both the system’s feedback (color-coded and numerical) and the textbox revision in which learners make the necessary changes to be submitted for further evaluation. As mentioned earlier, upon request, learners are offered a number of “Help Options” such as move/step definitions and examples, step statistics similar to what they automatically receive with respect to the moves in their discipline as part of the numerical feedback, access to the respective annotated corpus, and revision tips, all meant to assist them in their revision process.

Unique to IADE is the Encoding/Decoding module, which is a PHP module with a number of extensions that handles queries with the help of the database. It also receives student texts and transforms them into a format recognizable for processing. Then, it receives analyzed texts and numerical feedback from the Analysis module, transforms the texts into color-coded readable HTML format, and makes them available on the surface web interface. The Evidence Composite Library is the database that contains all the information that makes IADE functional. Specifically, it contains lists of:

1. User information (first and last name, login name, password, e-mail address)
2. User classification (graduate MA, graduate PhD, other)
3. Disciplines (50)
4. Annotated corpora (50 disciplines, 20 RA introductions each)

5. Students’ submitted drafts (automated analysis and numerical feedback)

6. Annotated introductions accessed by individual students (number of hits)

7. Steps and moves in annotated introductions accessed by individual students (number of hits)

8. Steps within each move

9. Step statistics (minimum, average, maximum percentage)

10. Help Options (Definitions, Step statistics, Annotated corpora, Revision tips) accessed by individual students (number of hits)

Lastly, the Response Processing is an analysis module, which incorporates two components: a Python script and an SVM classifier (see Pendar & Cotos, 2008). The script executes preprogrammed commands such as breaking the text into sentences and sending each sentence to the classifier to make the automated analysis possible. The classifier analyzes and classifies each sentence as belonging to a particular move, compares the move distribution in the student’s draft with annotated texts in the corpus of his/her discipline, and generates the feedback. It is always running in the background as a daemon, which allows concurrent access by multiple users. All these system components and their functions are technically more complicated, which can be seen in Appendix B.

**EVALUATION OF ICALL**

With the purpose of emphasizing the need for more grounded ICALL design, this paper demonstrates a complex way of developing a need driven and research and theory informed NLP-based application – IADE. Similar to other programs, e.g., CriterionSM, MyAccess!, WriteToLearn, etc., IADE provides remediation and offers students opportunities for guided practice. However, like other programs, IADE also needs to be evaluated in terms of its effectiveness for the intended purpose.

Despite the widely recognized promise of NLP technology, knowledge about the effectiveness of intelligent systems and their feedback for language learning is insufficient. Thus, not only are such systems often theoretically uninformed in their design, but they also have weak empirical support. To date, research on automated systems has largely been funded and carried out by the companies that have produced these commercial products. Most of it has focused on psychometric issues such as the validity and reliability of automated evaluation programs in the context of summative assessment (e.g., Burstein, 2003; Elliot, 2003; Keith, 2003; Landauer, Latham, & Foltz, 2003; Page, 2003). However, demonstrated reliability and validity in large-scale standardized testing contexts do not necessarily apply to instructional contexts. What is needed more now is an understanding of how these intelligent systems can contribute to...
Positive learning outcomes (Warschauer & Ware, 2006). “Programs like Criterion and MyAccessss! are relative newcomers to the field of second language writing, and their impact has yet to be systematically evaluated across a range of contexts, particularly those in which second language learners receive instruction,” explain Hyland and Hyland (2006, p.109). They argue that “the potential of automated essay evaluation for improving student writing is an empirical question, and virtually no peer-reviewed research has yet been published that examines students’ use of these programs or the outcomes” (p. 109).

Considering this paucity of research and the nature of existing empirical findings, Warschauer and Ware (2006) call for a new classroom research agenda that can help evaluate and guide the application of automated evaluation in the teaching of writing in order to make “maximal use of the software, or indeed, even for understanding if it ‘works’ in the broader sense of contributing to positive outcomes for student learning” (p. 10). Evaluating innovative technology, examples of which are IADE and other NLP-based systems, according to Chapelle (2007b), is “perhaps the most significant challenge teachers and curriculum developers face when attempting to introduce innovation into language education” (p. 30). Therefore, it is our task as applied linguists not only to design ICALL programs in a principled way, but to also analyze their effectiveness prior to practical classroom implementation. Consequently, the next stage after the development of IADE is to investigate its capability to enhance the formative assessment aspect of advanced-level academic writing instruction from multiple perspectives: language learning potential, meaning focus, learner fit, impact, authenticity, and practicality, all of which are criteria included in Chapelle’s (2001) CALL evaluation conceptual framework.

ACKNOWLEDGEMENTS

The author is extremely grateful to Nick Pendar for his work on the NLP component of IADE and to Andrew Vernon for his continuous work on the development, integration, and display of all the system components on the web as well as for his diagram included in Appendix B.

ENDNOTES

1These stages were earlier referred to as cognitive, associative, and autonomous (Fitts & Posner, 1967) and declarative, procedural, and automatic (Anderson, 1982).

2The database records the number of hits for further research purposes.

3Because the SVM model is very large, it is impractical to keep loading and unloading it
for each draft. Also, when more than one draft need to be processed at the same time, it is unfeasible to load more than one model at the same time. Because of this technical issue, the classifier was kept as part of the analysis module running as a background process, which in the Unix world is known as daemon. This allows concurrent analyses by the same model without the need to load and unload for each text submission.

REFERENCES


ICALL system featuring error contingent feedback. *Computer Assisted Language Learning, 17*(2), 177-201.


APPENDIX A

Move Schema Used by IADE for Automated Text Analysis

Move 1: Establishing a territory
Step 1 – Claiming centrality and/or
Step 2 – Making topic generalization/s and/or
Step 3 – Reviewing previous research

Move 2: Establishing a niche
Step 1A – Indicating a gap or
Step 1B – Highlighting a problem or
Step 1C – Question-raising or
Step 1D – Hypothesizing or
Step 1E – Adding to what is known or
Step 1F – Presenting justification

Move 3: Occupying the niche
Step 1A – Announcing present research descriptively or
Step 1B – Announcing present research purposefully
Step 2A – Presenting RQs and/or
Step 2B – Presenting hypotheses
Step 3 – Definitional clarifications and/or
Step 4 – Summarizing methods and/or
Step 5 – Announcing principal outcomes and/or
Step 6 – Stating the value of the present research and/or
Step 7 – Outlining the structure of the paper

(Based on Swales 1981, 1990, 2004)
APPENDIX B

IADE System Technical Architecture
APPENDIX C

List of Disciplines Annotated for and Represented by IADE

1. Accounting
2. Aerospace engineering
3. Agricultural and biosystems engineering
4. Agronomy
5. Analytical chemistry
6. Animal science
7. Applied linguistics
8. Architecture
9. Art and design
10. Bioinformatics
11. Biology
12. Biomedical Sciences
13. Biophysics
14. Business
15. Chemical engineering
16. Computer engineering
17. Computer science
18. Curriculum instruction
19. Database management
20. Economics
21. Electrical engineering and power systems
22. Electrophysiology
23. Environmental engineering
24. Food Science
25. Forest economy and management
26. Geological and atmospheric sciences
27. Genetics
28. Health human performance
29. Immunobiology
30. Industrial engineering
31. Inorganic/organic chemistry
32. Journalism
33. Mathematics
34. Materials science and engineering
35. Mechanical engineering
36. Meteorology
37. Microbiology
38. Molecular biology
39. Nano-scale and heat transfer
40. Physics and astronomy
41. Plant breeding
42. Plant physiology
43. Power systems economics
44. Public administration
45. Sociology
46. Special education
47. Statistics
48. Toxicology
49. Urban regional planning
50. Veterinary medicine