The Computer and Natural Language (Ling 245)

Special Topic: Grammatical Error Detection

Ross Israel
Dept. of Linguistics, Indiana University
2/13/2012
What Is Grammatical Error Detection?

We will be talking about errors made by learners in a second language acquisition context.

Language Learners often make non-native-like mistakes when constructing sentences:

- We arrived *to* the station.
- There is *the* garden in my house.
- *I eat* rice, nikujaga and salada yesterday.

Grammatical Error Detection entails trying to find these mistakes automatically.
Where Is This Useful?

- Automatic grading
  - Language teachers
  - Standardized testing
- Analysis and annotation of learner data for research
- Language learning software (ICALL)
CALL: Computer Assisted Language Learning
- Using computers and media in language learning and teaching
- Rosetta Stone, eLanguage
- Exercises are typically very simple in design, and offer little feedback
Where We Come In

- **CALL: Computer Assisted Language Learning**
  - Using computers and media in language learning and teaching
  - Rosetta Stone, eLanguage
  - Exercises are typically very simple in design, and offer little feedback

- **ICALL: Intelligent Computer Assisted Language Learning**
  - Utilize computational linguistics tools, such as POS tagging and Parsing along with statistical language modeling strategies
  - But, these tools often need to be altered to expect and diagnose errors, or at least handle learner data better
Most students in college or grad school have taken ETS tests:
  - GRE, TOEFL, SAT
Also:
  - TOEIC, PRAXIS, and a bunch more
*Criterion* is an English language ICALL system developed by ETS to assist teachers and students of English
Most of their products feature some level of NLP involvement
  - to foster language learning
  - to help with assessment
What is an Error?

- For things like the wrong word choice or a missing word, it's fairly easy to say that there is an error
  - *I need to go to store.*
- But, sometimes it's difficult to categorize an anomaly as an error
- The most frequent errors in English writing involve comma usage
  - These may be mechanical or grammatical, i.e. they might not necessarily indicate misunderstanding of a grammatical rule
  - *I saw, that you weren’t home.*
- What about spelling errors?
  - typos - *alphabrt, alfabet*
  - misuse of morphology - *drinked*
Learner Errors

Learners typically make different kinds of mistakes than native speakers.

- **Content Word Choice**
  - We need to deliver the merchandise on a daily *base/basis.

- **Preposition Error**
  - Our society is developing *in/at high speed.

- **Determiner Error**
  - There is *the/a garden in my house.
Commonly Used Techniques

- **Language Model - Gamon et al. (2008)**
  - Build n-grams (groups of consecutive words) of words, POS, and/or parse labels from native text and check if learner data n-grams align with the model we build

- **Web-based methods - Gamon and Leacock (2010)**
  - Take a few words of context on either side of a preposition to generate a web query
  - Replace the preposition with neighbors from a confusion set and search those queries
  - The search with the greatest number of hits is selected as the right answer

- **Heuristic-based systems - Eeg-Olofsson and Knutsson (2003)**
  - Write linguistic rules designed to find errors in learner data

- **Statistical - Tetreault and Chodorow (2008)**
  - Statistical methods means building a classifier
  - So, what is a classifier?
Understanding Classifiers

- Classifiers are a typical tool used in *machine learning*.
  - Machine learning is not as scary as it sounds!
  - Software learns to predict outcomes based on previous events.

- We need two sets of data:
  - Training Set - needs to be big
  - Testing Set - usually smaller

- The data sets are full of events (*instances*) that contain *features* that describe the circumstances of the event and a *class* that is the answer we are trying to guess.
Introduction

NLP and (I)CALL

Types of Errors

Error Detection in Action

Techniques

A Quick Intro to Machine Learning

English Prepositions
Comma Error Detection
Comma Uses
Error Detection
Korean Particles
Classification

References

Machine Learning Basics

Test Data
Learner Essays

Training Data
Well-edited text

Annotators

Annotated “Gold” Test Data

Model

Results

Made with lovelycharts.com
Selecting Features

Let’s consider a real-world example:

- The Task: We want to classify the weather as either good or bad.
- We would want features like
  - temperature
  - sunny?
  - cloudy?
  - windy?
  - humidity level
  - rain/snow/none
Running the Classifier

- Then, we would build vectors for every measurement we take and *label* them to build training data:
  - 75, yes, no, no, 70%, none, *good*
  - 35, no, no, yes, 50%, none, *bad*
  - 105, yes, no, no, 98%, rain, *bad*
  - 68, yes, yes, no, 75%, none, *good*

- Now, when we give the classifier an unknown feature vector, we hope that it makes a wise decision
Running the Classifier

- Then, we would build vectors for every measurement we take and *label* them to build training data:
  - 75, yes, no, no, 70%, none, good
  - 35, no, no, yes, 50%, none, bad
  - 105, yes, no, no, 98%, rain, bad
  - 68, yes, yes, no, 75%, none, good

- Now, when we give the classifier an unknown feature vector, we hope that it makes a wise decision
  - 85, yes, no, no, 65%, none - classifier’s guess = *good* yay!
Running the Classifier

- Then, we would build vectors for every measurement we take and *label* them to build training data:
  - 75, yes, no, no, 70%, none, *good*
  - 35, no, no, yes, 50%, none, *bad*
  - 105, yes, no, no, 98%, rain, *bad*
  - 68, yes, yes, no, 75%, none, *good*

- Now, when we give the classifier an unknown feature vector, we hope that it makes a wise decision
  - 85, yes, no, no, 65%, none - classifier’s guess = *good* yay!
  - 15, no, no, yes, 70%, snow - classifier’s guess = *bad* yay!
Running the Classifier

- Then, we would build vectors for every measurement we take and *label* them to build training data:
  - 75, yes, no, no, 70%, none, good
  - 35, no, no, yes, 50%, none, bad
  - 105, yes, no, no, 98%, rain, bad
  - 68, yes, yes, no, 75%, none, good

- Now, when we give the classifier an unknown feature vector, we hope that it makes a wise decision
  - 85, yes, no, no, 65%, none - classifier’s guess = good yay!
  - 15, no, no, yes, 70%, snow - classifier’s guess = bad yay!
  - 75, yes, no, yes, 70%, none - classifier’s guess = bad oops!

- Evaluation metrics:
  - Accuracy - # correct / total
  - Precision - when we say “good” how often are we right?
  - Recall - how many “good” instances do we find?
  - F-score - combination of the two
Theoretical Interlude

- Recall and Precision *usually* have an inverse relationship
  - The higher the recall, the lower the precision, and vice versa
- We can think about recall and precision in lots of real world scenarios
  - Sports referees
  - Airport security
  - Search engines
- For error detection, do we want better recall or precision?
Theoretical Interlude

- Recall and Precision *usually* have an inverse relationship
  - The higher the recall, the lower the precision, and vice versa
- We can think about recall and precision in lots of real world scenarios
  - Sports referees
  - Airport security
  - Search engines
- For error detection, do we want better recall or precision?
- Focus on precision; we don’t want to tell a learner that they’ve made a mistake when they haven’t!
Why Prepositions?

Some common areas of research in English error detection are articles, prepositions, and collocations. We’ll look a little more in depth at prepositions.
Why Prepositions?

Some common areas of research in English error detection are articles, prepositions, and collocations. We’ll look a little more in depth at prepositions.

- Because prepositions make up a large portion of errors commonly made by learners, there has been a good deal of research on how to find and diagnose preposition errors.
- Also, prepositions are a closed set, so it’s a problem that’s easier to define than a more open error type like use of the wrong content word.
- Prepositions can be treated as a confusion set where we know that one is being substituted for another.
Choosing the correct preposition can be a tough task even for native speakers.
Choosing the correct preposition can be a tough task even for native speakers.

There has been concern about syncing phone contacts to Facebook. “As long as you are aware of who is in the group, it can be a great privacy tool. If it gets out of hand, it could give you a sense of false security.” The rollout of new products comes with reports that a syncing feature allows Facebook access to contact data and share it to the site. “It’s very possible that your private phone numbers - and those of lots of your and their friends - are on the site,” said Charles Arthur from the Guardian newspaper.
Choosing the correct preposition can be a tough task even for native speakers.

There has been concern over syncing phone contacts with Facebook. "As long as you are aware of who is in the group it can be a great privacy tool. If it gets out of hand it could give you a sense of false security." The roll out of new products comes amid reports that a syncing feature on the iPhone lets Facebook access contact data and share it on the site. "It’s very possible that your private phone numbers - and those of lots of your and their friends - are on the site," said Charles Arthur of the Guardian newspaper.
Machine Learning for Prepositions

- Tetreault and Chodorow used a maximum entropy classifier to try to find preposition confusions and extraneous uses.

- They extracted 25 features including:
  - words/POS tags in a 2 word window(+/-) around preposition
  - the head verb and noun of the preceding VP and NP
  - the head noun of the following NP

Tetreault and Chodorow used a maximum entropy classifier to try to find preposition confusions and extraneous uses. They extracted 25 features including:

- words/POS tags in a 2 word window(+/-) around preposition
- the head verb and noun of the preceding VP and NP
- the head noun of the following NP

Machine Learning for Prepositions

- Tetreault and Chodorow used a maximum entropy classifier to try to find preposition confusions and extraneous uses.

- They extracted 25 features including:
  - words/POS tags in a 2 word window(+/-) around preposition
  - the head verb and noun of the preceding VP and NP
  - the head noun of the following NP

Tetreault and Chodorow used a maximum entropy classifier to try to find preposition confusions and extraneous uses. They extracted 25 features including:

- words/POS tags in a 2 word window(+/-) around preposition
- the head verb and noun of the preceding VP and NP
- the head noun of the following NP
Tetreault and Chodorow used a maximum entropy classifier to try to find preposition confusions and extraneous uses.

They extracted 25 features including:
- words/POS tags in a 2 word window (+/-) around preposition
- the head verb and noun of the preceding VP and NP
- the head noun of the following NP

John went to the store this morning.

- word+POS bigrams: went_VBD, the_DET
- head of previous VP = went
- head of previous NP = John

Their system achieved 84% precision and 19% recall.
- This might sound low, but keep in mind, we want to get the best possible precision, even if it means losing recall.
Motivation

Why should we bother detecting comma errors?

- Commas account for a significant proportion of errors among both native and non-native writers
  - 6 of the top 20 errors for native writers
  - 4th most common error among non-native writers
- Incorrect sentence internal comma placement leads to ambiguity and/or unintelligibility
- Teachers have asked ETS for this kind of error detection in Criterion
- There has not, as far as we know, been any work published specifically on detecting comma errors in writing
  - However, comma restoration has been actively researched for a number of years, especially in the ASR community
TOEFL Errors

- Learners make a variety of errors with commas
  - both omission and extraneous usage

  *If you want to be a master you should know your subject well.*

  *This type of situation cost money and time and that is the most important idea for many companies around the world*

  *I suppose, that it is better to specialize in one specific subject.*

- Detecting, diagnosing, and correcting these errors will be useful in learning and assessment environments
Collecting Comma Rules

- We compiled a comprehensive list of comma usage and made a list of 15 common comma uses
  - Elements in a list
    
    *Paul put the kettle on, Don fetched the teapot, and I made tea.*
  - Introductory words and phrases
    
    *Hopefully, this car will last for a while.*
  - Dependent Clauses
    
    *After I brushed the cat, I lint-rollered my clothes.*
  - Independent Clauses
    
    *I have finished painting, but he is still sanding the doors.*
  - Parentheticals
    
    *My father, a jaded and bitter man, ate the muffin.*

Also: Quotations, Adjectives, Conjunctive Adverbs, Contrasting Elements, Number, Dates, Places, Titles, List introducers, Other
We use the comma rules to develop an annotation scheme to apply to 60 TOEFL essays and 60 *Criterion* essays (10 from each Score Point).

- Scores are awarded 1-6 for writing quality

- We can use the annotated data to evaluate our error detection system’s performance

- We found that:
  - 85% of writers’ existing commas are correct
  - More often, commas are omitted
  - The 5 most frequent categories account for over 80% of all commas
    1. Introductory words/phrases: 39%
    2. Parentheticals: 18%
    3. Lists: 11%
    4. Dependent clauses: 11%
    5. Independent clauses: 4%
Inside the System

- Test on our annotated TOEFL and Criterion essays
- First, we remove every comma from the text
- Every space between words in a text is a candidate for comma insertion
- The classifier makes decisions based on a training model of gold text (i.e., all commas are correct)
- We use GRE essays to stay within the essay domain for training our system
  - We use only essays with a ‘6’ rating from human scorers
- System features:
  - 1,2,3-grams of words and POS tags in a 5 word window
  - combination (word+POS) features in that same window
  - Distance: From BOS, to EOS, from previous CC, to next CC
  - First word+pos of the sentence
- We only guess errors when the system is 90% confident
The Computer and Natural Language
Grammatical Error Detection

Introduction
NLP and (I)CALL
Types of Errors
Error Detection in Action
Techniques
A Quick Intro to Machine Learning
English Prepositions
Comma Error Detection
Comma Uses

Data | Acc. | P  | R  | F  | n  |
-----|------|----|----|----|----|
TOEFL | 98.3 | 94.0 | 31.7 | 47.4 | 297 |
Criterion | 97.8 | 84.9 | 20.0 | 32.4 | 365 |
Combined | 98.1 | 89.8 | 25.2 | 39.4 | 662 |

* Precision - How often do we correctly predict an error?
* Recall - What percentage of errors do we really find?
* F-score - Combination of Precision and Recall

Data Acc. P R F n
TOEFL 98.3 94.0 31.7 47.4 297
Criterion 97.8 84.9 20.0 32.4 365
Combined 98.1 89.8 25.2 39.4 662
Output Analysis

For research in any particular topic, you have to ...
- Learner did not use a comma
- 61% confidence for No comma

For example, if you specialize in English literature of the 19th century than you probably ...
- Learner used a comma
- 99% confidence for No comma

A PhD doctor of chemistry who has knowledge only about chemistry cannot participate in a talk program, which includes thought about how to make the environment a better place to live.
- Learner did not use a comma
- 92% confidence for a comma
Korean language learning

- This is based on work that Markus and I have been doing with Sun-Hee Lee - Dickinson et al. (2010)
- Ultimate goal: develop computational tools to assist learners of Korean
Korean language learning

- This is based on work that Markus and I have been doing with Sun-Hee Lee - Dickinson et al. (2010)
- Ultimate goal: develop computational tools to assist learners of Korean
- Korean has a number of features uncommon to Western languages:
  - agglutinative morphology, rich system of case marking, relatively free word order
This is based on work that Markus and I have been doing with Sun-Hee Lee - Dickinson et al. (2010)

Ultimate goal: develop computational tools to assist learners of Korean

Korean has a number of features uncommon to Western languages:
  - agglutinative morphology, rich system of case marking, relatively free word order
  - One prominent area of difficulty for learners is that of post-positional particles
Korean language learning

- This is based on work that Markus and I have been doing with Sun-Hee Lee - Dickinson et al. (2010)
- Ultimate goal: develop computational tools to assist learners of Korean
- Korean has a number of features uncommon to Western languages:
  - agglutinative morphology, rich system of case marking, relatively free word order
  - One prominent area of difficulty for learners is that of post-positional particles

We want to build a machine learner to detect errors in particles
Background: Korean particles

- Similar to English prepositions, but wider range of functions:

  - Case marker/Semantic role markers: (1) Sumi-ka

    Sumi-SBJ
    John-eykey
    book-OBJ
    read-polite

    'Sumi reads a book to John.'

  - Modifiers (cf. prepositions): indicate specific lexical, syntactic, & semantic information between verb & noun (2) Sumi-ka

    Sumi-SBJ
    John-uy
    house-LOC
    ku-lul
    he-OBJ
    twu
    two
    sikan-ul
    hours-OBJ
    kitaly-ess-ta.

    'Sumi waited for John for (the whole) two hours in his house.'
Background: Korean particles

- Similar to English prepositions, but wider range of functions:
  - Case marker/Semantic role markers:

(1) Sumi-**ka** John-**eykey** chayk-**ul** ilhke-yo
Sumi-SBJ John-to book-OBJ read-polite

‘Sumi reads a book to John.’
Background: Korean particles

- Similar to English prepositions, but wider range of functions:
  - Case marker/Semantic role markers:
    (1) Sumi-\textbf{ka} John-\textbf{eykey} chayk-\textbf{ul} ilhke-yo
        Sumi-SBJ John-to book-OBJ read-polite
        ‘Sumi reads a book to John.’
  - Modifiers (cf. prepositions): indicate specific lexical, syntactic, & semantic information between verb & noun
    (2) Sumi-\textbf{ka} John-\textbf{uy} cip-\textbf{eyse} ku-lul twu
        Sumi-SBJ John-GEN house-LOC he-OBJ two
        sikan-\textbf{ul} kitaly-ess-ta.
        hours-OBJ wait-PAST-END
        ‘Sumi waited for John for (the whole) two hours in his house.’
Korean particles: expected errors

Learners of Korean often misuse particles

(3) a. Sumi-

\textit{nun} chayk-\textit{i} philyohay-yo
Sumi-TOP book-SBJ need-polite
‘Sumi needs a book.’

\textit{nun}
Korean particles: expected errors

Learners of Korean often misuse particles

(3) a. Sumi-\textit{nun} chayk-\textit{i} philyohay-yo
   Sumi-TOP book-SBJ need-polite
   ‘Sumi needs a book.’

b. *Sumi-nun chayk-\textbf{ul} philyohay-yo
   Sumi-TOP book-OBJ need-polite
   ‘Sumi needs a book.’
Korean particles: expected errors

Learners of Korean often misuse particles

(3) a. Sumi-*nun* chayk-*i* philyohay-yo
    Sumi-TOP book-SBJ need-polite
   ‘Sumi needs a book.’

   b. *Sumi-nun chayk-*ul* philyohay-yo
    Sumi-TOP book-OBJ need-polite
   ‘Sumi needs a book.’

particle errors by learners of Korean can be categorized into 6 types:
- omission, replacement, addition, malformation, paraphrasing, and spacing
Machine learning paradigm

Parallel errors made by ESL learners for prepositions

- We can base our system on the work by Tetreault and Chodorow, but we need to consider the differences between English and Korean
Parallel errors made by ESL learners for prepositions

- We can base our system on the work by Tetreault and Chodorow, but we need to consider the differences between English and Korean

Some major differences in Korean:

- Particles are post-positional - they show up after a word instead of before it like in English
- Base word order is SOV
  - Need to look at following verb & following noun
- Morphological composition of words is different
  - Agglutinative: stem + suffixes
Our Features

- The feature vector is built on a five word window that includes the target word and two words (+/-) for context.
Our Features

- The feature vector is built on a five word window that includes the target word and two words (+/-) for context.

- Each word is broken down into four features:
  - Stem, affixes, stem_POS, affixes_POS
    - Use trigram+rule based morphological tagger for Korean
  - Include features for preceding and following nouns & verbs (roots only)

- If the target word affixes contains a particle, it is removed and used as the class; otherwise the class is NONE.
An Example Instance

  use-Past-Decl
  ‘While living in America, (I/she/he) used only English at home.’

b. Mikwuk NPR NONE NONE
  sal VV myense ECS
  Yenge NPR NONE NONE
  cip NNC NONE NONE
  ss VV ess+eyo EPF+EFN
  sal Mikwuk ss cip
  man-ul
Results

- Our system was able to achieve around 45% precision and around 18% recall for this task.
- These results are low, but
  - This task is really hard!
  - We have been focusing on preliminary issues to this point (e.g. data collection and annotation)
  - Now we are working on improving these results.
References


