Automated Writing Assistance: Grammar Checking and Beyond

Topic 3: Grammar Checking

Robert Dale
Centre for Language Technology
Macquarie University
The Need

Grammar Checker

You might have asked yourself before sending an important email to a business colleague or a new friend:

"Will this text read better if I perform a grammar check?"

You are not alone! People all around the world find themselves asking this question when trying to avoid grammar mistakes in their texts.

Grammar Checker – The Ultimate Solution for Your Grammar Errors

Proofread your text in a single click by using an online grammar checker. An online grammar checker will save you the embarrassment of sending a text with grammar mistakes and will make your text look more professional and reliable.
Outline

• What is a Grammatical Error?
• Grammar Checking without Syntax
• IBM’s EPISTLE
• Grammar Checking Techniques
• Related Areas
• Commercial Packages
What is a Grammatical Error?

- Something that breaks the rules of the language
- Who decides?
  - Dialects
  - Formality
  - Language change
- Some jurisdictions are stricter than others
  - L'Académie française and its 40 ‘immortals’
Agreement Errors: The Paradigm Grammatical Error

- John and Mary is coming today.
- A blocks are red.
Taxonomies of Error: Douglas and Dale 1991

- Spelling Errors
- Syntactic Errors
  - Semantic Problems
  - Stylistic Problems
  - Rhetorical Problems
  - Punctuation Problems

- Co-occurrence Errors
- Dependency and Subordination Errors
  - Number Disagreement
  - Bad Subcategorisation
  - Resumptive Pronoun
  - Syntactic Parallelism
  - Bad Clause Conjunction
  - Misleading PP Attachment
  - Misleading Adverbial Attachment
  - Missing Subordination Indicator
  - Redundancy
Subject–Verb Number Disagreement

• But the males in this study experienced significant difficulties in this area and this problem suggest that some more attention be paid to the phenomenon.
• This method requires a user to think aloud while performing a task, while the researchers makes notes, and perhaps records the session on audio or video tape.
• The main reported problems was the Unix editor vi.
Subject–Verb Number Disagreement

• But the males in this study experienced significant difficulties in this area and this problem suggest that some more attention be paid to the phenomenon.

• This method requires a user to think aloud while performing a task, while the researchers makes notes, and perhaps records the session on audio or video tape.

• The main reported problems was the Unix editor vi.

→ The main reported problems were with the Unix editor vi.
Incorrect Subcategorisation Frames: Verbs

- Both Carroll’s work and our own, however, has tended to use existing commercial manuals as a basis --- and the question then is how to prune to a fraction of their original size, and to alter their contents to approach more closely to the problems that users actually confront when trying to learn a new system.
Incorrect Subcategorisation Frames: Verbs

- Both Carroll’s work and our own, however, has tended to use existing commercial manuals as a basis --- and the question then is how to prune to a fraction of their original size, and to alter their contents to approach more closely to the problems that users actually confront when trying to learn a new system.
Incorrect Subcategorisation Frames: Nouns and Prepositions

- Their feedback pointed to problem areas and causes for misinterpretation, and suggestions of improvements offered by them.
Incorrect Subcategorisation Frames: Nouns and Prepositions

- Their feedback pointed to problem areas and causes for misinterpretation, and suggestions of improvements offered by them.

→ Their feedback pointed to problem areas and causes of misinterpretation, and suggestions for improvements offered by them.
Incorrect Subcategorisation Frames: Verbs

• In this way, it is anticipated that the issue of native users not really knowing what it is they need to know is dealt with.
Incorrect Subcategorisation Frames: Verbs

• In this way, it is anticipated that the issue of native users not really knowing what it is they need to know is dealt with.

→ In this way, it is anticipated that the issue of native users not really knowing what it is they need to know will be dealt with.
Incorrect Subcategorisation Frames: Nouns and Prepositions

- All mailing systems have capabilities of composing, sending and receiving messages.
Incorrect Subcategorisation Frames: Nouns and Prepositions

- All mailing systems have **capabilities of** composing, sending and receiving messages.

→ All mailing systems have **facilities for** composing, sending and receiving messages.
Incorrect Subcategorisation Frames: Adjectival Complements

- The feature checklist was easy to administer and complete by experienced users ...
Incorrect Subcategorisation Frames: Adjectival Complements

- The feature checklist was easy to administer and complete by experienced users ...

→ The feature checklist was easy to administer and easy for experienced users to complete ...
Syntactic Parallelism Failures

- Semi-structured interviews were conducted with experienced users to find what their most common tasks, the tasks a new user would need to begin, and what errors would be most likely in the early stages.
Semi-structured interviews were conducted with experienced users to find what their most common tasks were, what tasks a new user would need to begin, and what errors would be most likely in the early stages.
Bad Clause Conjunction

- It had approximately 13% of the pages of the commercial manual, it allowed 30% faster learning and more effective use of the email system overall, and significantly better performance on individual subtasks including recovery from error.
Bad Clause Conjunction

- It had approximately 13% of the pages of the commercial manual, it allowed 30% faster learning and more effective use of the email system overall, and significantly better performance on individual subtasks including recovery from error.

→ It had approximately 13% of the pages of the commercial manual, it allowed 30% faster learning and more effective use of the email system overall, and it gave significantly better performance on individual subtasks including recovery from error.
Bad Clause Conjunction

• The conditions under which our subjects worked tended to minimize such problems — since we asked them to persevere, and in the end they would be able to get human help.
Bad Clause Conjunction

• The conditions under which our subjects worked tended to minimize such problems — since we asked them to persevere, and in the end they would be able to get human help.

→ The conditions under which our subjects worked tended to minimize such problems, since we asked them to persevere, and in the end they would be able to get human help.
Bad Clause Conjunction

- The more active but ineffectual behaviour of the males may mean that they feel they must be capable of mastering the system, of overcoming their errors and are less worried or affected by the possibility of making errors.
Bad Clause Conjunction

- The more active but ineffectual behaviour of the males may mean that they feel they must be capable of mastering the system and of overcoming their errors and are less worried or affected by the possibility of making errors.

→ The more active but ineffectual behaviour of the males may mean that they feel they must be capable of mastering the system and of overcoming their errors, and are less worried or affected by the possibility of making errors.
Bad Clause Conjunction

• Novice users should, however, be able to voice thoughts and desires on any topic, throughout the process if the manual is to be properly user-centred.
Bad Clause Conjunction

- Novice users should, however, be able to voice thoughts and desires on any topic, throughout the process if the manual is to be properly user-centred.

→ However, if the manual is to be properly user-centred, novice users should be able to voice thoughts and desires on any topic throughout the process.
Syntactic Redundancy

- So although this seems to be a winning feature in learning, it may not …
- … this problem suggests that some more attention be paid to the phenomenon
- … thus so this argues for the complementary use of …
Syntactic Redundancy

• So although this seems to be is a winning feature in learning, it may not …

• … this problem suggests that some more attention be paid to the phenomenon

• … thus so this argues for the complementary use of …
What Causes Grammar Errors?

• Competence-based errors:
  – Unfamiliarity with the language

• Performance-based errors:
  – Repeated words
  – Editing errors
Outline

• What is a Grammatical Error?
• Grammar Checking without Syntax
• IBM’s EPISTLE
• Grammar Checking Techniques
• Related Areas
• Commercial Packages
The Unix Writer’s Workbench

- A breakthrough in the early 1980s
  - We believe that the Writer's Workbench programs provide a more general text analysis system than JOURNALISM or CRES, and unlike EPISTLE they are already in wide use. At Bell Laboratories there are over 1000 users on over 50 machines. [1982:106]

- Widely-used in educational contexts

- Underlying technology formed the basis for the first PC grammar checkers: Grammatik, RightWriter, StyleWriter
The Unix Writer’s Workbench: Proofreading with PROOFR

• Checks for existence of non-word spelling errors; user-specified automatic correction can be carried out
• Checks for unbalanced punctuation and other simple punctuation mistakes
• Checks for double words
• Checks for misused words, wordy phrases, sexist terms, …
• Checks for split infinitives using a simple PoS tagger
The Unix Writer’s Workbench: Stylistic Analysis with STYLE

• Based on PoS tagging, provides 71 numbers describing stylistic features of the text
  – Readability indices
  – Average sentence and word length
  – Distribution of sentence lengths
  – Percentage of verbs in passive voice
  – Percentage of nouns that are nominalisations
  – …
The Unix Writer’s Workbench: Stylistic Analysis with STYLE

<table>
<thead>
<tr>
<th>readability grades: (Kincaid) 11.3 (auto) 12.6 (Coleman-Liau) 13.1 (Flesch) 13.2 (48.8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sentence info: no. sent 240 no. wds 4636</td>
</tr>
<tr>
<td>av sent leng 19.3 av word leng 5.18</td>
</tr>
<tr>
<td>no. questions 1 no. imperatives 0</td>
</tr>
<tr>
<td>no. content wds 2734 59.0% av leng 6.72</td>
</tr>
<tr>
<td>short sent (&lt;14) 24% (58) long sent (&gt;29) 9% (22)</td>
</tr>
<tr>
<td>longest sent 64 wds at sent 150; shortest sent 4 wds at sent 70</td>
</tr>
<tr>
<td>sentence types: simple 42% (101) complex 38% (92)</td>
</tr>
<tr>
<td>compound 7% (16) compound-complex 13% (31)</td>
</tr>
<tr>
<td>word usage: verb types as % of total verbs</td>
</tr>
<tr>
<td>to be 32% (170) aux 16% (85) inf 17% (89)</td>
</tr>
<tr>
<td>passives as % of non-inf verbs 14% (63)</td>
</tr>
<tr>
<td>types as % of total</td>
</tr>
<tr>
<td>prep 10.5% (487) conj 3.8% (177) adv 4.2% (197)</td>
</tr>
<tr>
<td>noun 28.0% (1296) adj 17.2% (797) pron 4.7% (220)</td>
</tr>
<tr>
<td>nominalizations 2% (90)</td>
</tr>
<tr>
<td>sentence beginnings: subject opener: noun (48) pron (28) pos (1) adj (35) art (57) tot 70%</td>
</tr>
<tr>
<td>prep 13% (32) adv 6% (15)</td>
</tr>
<tr>
<td>verb 1% (3) sub_conj 6% (14) conj 2% (5)</td>
</tr>
<tr>
<td>expletives 1% (2)</td>
</tr>
</tbody>
</table>
The Unix Writer’s Workbench: Other Components

- **PROSE:** compares the stylistic parameters of a given text against a domain-specific standard
- **ABST:** determines the conceptual abstractness of a text via a list of 314 abstract words
- **ORG:** prints only first and last sentences of paragraphs
Atwell [1987]: CLAWS

- Originally built to assign PoS tags to the London-Oslo-Bergen corpus
- Developed in part because of the computational cost of more complex systems:
  - ‘[Heidorn et al 82] reported that the EPISTLE system required a 4Mb virtual machine (although a more efficient implementation under development should require less memory).’ [1987:38]
Atwell [1987]:
Constituent-Likelihood Error Detection

- For PoS tagging, uses a table of PoS bigram frequencies to determine most likely sequences
- Detects grammatical errors by flagging unlikely PoS transitions
- Doesn’t need separate data for training error likelihoods
Outline

• What is a Grammatical Error?
• Grammar Checking without Syntax
  • IBM’s EPISTLE
• Grammar Checking Techniques
• Related Areas
• Commercial Packages
IBM’s EPISTLE: History

- Initial work in the early 1980s led to several innovative techniques
- Based on Heidorn’s Augmented Phrase Structure Grammar [1975]
- Renamed CRITIQUE somewhere in the mid to late 1980s
- Released on IBM mainframes late 1980s
- Key team members went on to build Microsoft Word’s grammar checker from 1992 onwards
- Grammar checking released as part of MS Word 97
IBM’s CRITIQUE: Grammar vs Style

• Grammatical critiques:
  – Strict rules as to whether a sentence is grammatical or not
  – Correction is typically clear

• Stylistic weaknesses are less black and white:
  – too great a distance between subject and verb
  – too much embedding
  – unbalanced subject/predicate size
  – excessive negation or quantification
  – …
IBM’s CRITIQUE:
Grammar Errors

- **Number Disagreement:**
  - he go, many book, it clarifies and enforce
- **Wrong Pronoun Case:**
  - between you and I, it is me
- **Wrong Verb Form:**
  - had expect, seems to been
- **Punctuation:**
  - run-on sentences, questions with a final period instead of a question mark
- **Confusions:**
  - who’s vs whose, it’s vs its, your vs you’re, form vs from
IBM’s CRITIQUE: Stylistic Weaknesses #1

- Excessive length
  - Sentences or lists that are too long
  - Sequences with too many prepositional phrases
- Excessive complexity
  - Noun phrases with too many premodifiers
  - Clauses with a series of *ands*
  - Verb phrases with too many auxiliary verbs
  - Clauses with too much negation
- Lack of parallelism
  - Example: you should drink coffee rather than drinking tea
IBM’s CRITIQUE:
Stylistic Weaknesses #2

- Excessive formality
  - phrases that are bureaucratic, pompous or too formal
- Excessive informality
  - constructions acceptable in spoken English but too informal when written
- Redundancy
  - phrases that can be shortened without loss in meaning
- Missing punctuation
- Nonpreferred constructions
  - Split infinitives [eg to completely remove], colloquial usage [eg ain’t working]
The MS Word Grammar Checker: Processing Steps

1. Tokenisation and Lexical Lookup
2. Syntactic Sketch
3. Syntactic Portrait
4. Production of Logical Forms
The MS Word Grammar Checker: An Example

- Consider the following sentence:
  - After running a mile he seemed tired.
The MS Word Grammar Checker: Lexical PoS Records

- Also includes detection of multiword elements and named entity mentions
- Lexicon based on LDOCE and AHD + supplementary information added both manually and automatically
- Over 100k words
- There are two other records produced for ‘after’ here for the Adj and Adv uses

---

```plaintext
{Segtype  PREP
 Nodetype  PREP
 Nodename  PREP1
 Ft-Lt     1-1
 String    "After"
 CopyOf    REC40
 Lex       "After"
 Lemma     "after"
 Bits      TakesAn InitCap Tme
 Prob      1.00000 }

{Segtype  CONJ
 Nodetype  CONJ
 Nodename  CONJ1
 Ft-Lt     1-1
 String    "After"
 CopyOf    REC41
 Lex       "After"
 Lemma     "after"
 Bits      Subconj TakesAn
 InitCap Tme
 Prob      0.00119 }
```
The MS Word Grammar Checker: Syntactic Analysis

- Bottom-up chart parser
- Uses probabilities and heuristics
- Grammar contains 125 mostly binary rules
- This is the derivation tree
The MS Word Grammar Checker: Syntactic Analysis

After running a mile he seemed tired.

---

SSLST 2011
The MS Word Grammar Checker: Syntactic Information Stored at the Root Node
The MS Word Grammar Checker:
The VP→ VP PP Rule [Abbreviated]

VPwPP:

```
PP ( ^Comma(Prp) & ^Nappcomma(lastrec) & ^Precomma(lastrec) & ^SuspSUBCL & (forany(Prmods, [Comma]) -> Coords) &
  forall(lastrecs, [Comma | Digits^=3 & Digits^=4]) & (forany(lastrecs, [Comma & ^Paren]) -> (Multcomma | Comma(lastphr))) &
  forall(lastrecs, [Noncomp | ^T5 | (Compl & Lemma(lasttokn)^="that")]) & (Gerund -> (Rel(Postadv) | Postadv^=lastrec)) &
  Lemma(Prp) ^in? set{a an but x X} & forall(Coords, [Lemma(Prp) ^in? set{a an but x X}]) )
```

```
VP ( ^Semiaux & ^Relpn & ^Paren &
  (forany(lastrecs(PP), [Nappcomma]) -> (^Pastpart | ^PPobj(first(Prmods)) |
    ^Comma(Prp)))) &
  forall(lastrecs(PP), [Nappcomma -> (AMultcomma | Numbr ^Agree? Numbr(VP))]) &
  (Nodetype(lastrec(PP))="RELCL" -> (Atthatcomp(lasttokn(PP)) |
    Rel(first(Prmods(lastrec(PP)))))) & Nodetype(last(Prmods(PP))) ^in? set{SREL, TAG} &
  (Ord(Adj(Lasttokn(PP)))) -> (Adj(Lasttokn(PP))) & ^Numbr(Adj(Lasttokn(PP))) ) &
  (Adv(Lex) -> (Prmods | Obj1 | (^Confus & Lemma ^in? set{no yes}))) & (Wh(Conj(Lex(PP))) -> (Prmods(PPobj(PP)) | YNQ)) &
  (Digits(first(Prmods)) -> (^Comma(first(Prmods)) | Prmods(first(Prmods)) | Nodetype(lasttokn(PP))="NOUN")) &
  (Mnth(lasttokn(PP)) -> (^Ord(firsttokn) | ^Digits(firsttokn) | Digits(firsttokn)>2)) &
  (Nom(Prn(Prp))) ) & (^Obj(Prn(Lex(PP)))) )
```

```sql
-- VP { Prmods=PP+Prmods; Props=Props(PP)+Props; ^SuspNREL;
  if (Subject(PP) ^in? Prmods(PP) & FortoPP(PP)) {Subject=PP; -VPInvert;}
  else if (!Subject(PP) | VPInvert(VP) & ^thesubj_test(VP)) MidPPs=PP+MidPPs; 
  else {TopPPs=PP+TopPPs; Inverts=PP+Inverts;}; Pod=Pod+Pod(PP); 
  if (Lemma(lasttokn(PP))==";"); Pod=Pod-4; 
  if (PPobj(PP) & Loc(Adv(Lex(PP)))) Pod=Pod-1; 
  if (Subject in? Prmods(PP) | theresubj_test) Pod=Pod+1; }
```
The MS Word Grammar Checker: A Logical Form
The MS Word Grammar Checker:
An Error Checking Rule

Desc_Comma5:

SYNREC (((Nodetype in? set{SUBCL AVP PRPRTCL AVPNP INFCL}) |
  (Nodetype=="PP" & PObj)) &
  seg==first(Prmods(Parent)) &
  Nodetype(lasttokn) ^= "CHAR" &
  ^Theresubj &
  seg ^= Subject(Parent) &
  (Nodetype=="AVP" -> (^TheAVP & ^forany(Prmods,[TheAVP]))) &
  (Wh -> Lemma=="however") &
  ^forany(Coords,[Wh]) &
  (Nodetype(Head(Parent))=="VERB" | VPcoord(Parent)) &
  (Neg -> ^YNQ(Parent)) &
  ((Subject(Parent) &
    ((Ft(Subject(Parent))<Ft(FrstV(Parent)) & Ft(Subject(Parent))>Ft) |
    (VPcoord(Parent) & Ft(Subject)<Ft(FrstV(first(Coords(Parent))))) |
    Nodetype(Parent)=="IMPR" |
    (Nodetype(Parent)=="QUES" & (YNQ(Parent) | WhQ(Parent))))))

--> SYNREC { { segrec rec, commarec;
    commarec=segrec(Nodetype="CHAR"; Lemma="","};
    rec=segrec(%%SYNREC; Psmods=Psmods++commarec};
    add_descrip("Comma with Adverbials",0,rec); } }
The MS Word Grammar Checker: A Segment Record with An Error

```plaintext
>display record PP1
(SegType PP
NodeType PP
NodeName PP1
Ft-Lt 1-4
String "After running a mile"
CopyFE NP3
RuleS [TrLP_ControlatVP Desc_CommaS NPtoPP PRPRCLtoNP PPtoPRCL VPwNP1 VERBtoVP]
ConstitS [PP1 PP1 NP3 NP3]
Law "running"
Lemma "run"
Bits Prep3 Sing L3 X9 Wv6
I0 D1 T1 L1 L7 T5
AsubJ Loc_gr Unacc Mov Middle Wv4
Prob 0.88383
Primads PP2 "After"
Head VERB1 "running"
Primads NP1 "a mile"
Gerund VERB1 "running"
PpObj NP3 "running a mile"
Prp PP2 "After"
Obj1 NP1 "a mile"
Pres PPPRCL "running a mile"
Pod 1d
Parent DECL1 "after running a mile he seemed tired ."
Nargs 1
FrcType VERB3 "running"
Object NP1 "a mile"
VpOb [along around away back down in off on out over through up across after]
VpPrp [across after at from with over into on of through to against]
Descr (Ft-Lt 1-4
Value 18
DescrType "Comma with Adverbials"
DescrRep1 PP4 "After running a mile"
DescrRep1Str "After running a mile,"
ScanNode run1
PrevVerb PP ]
```
The MS Word Grammar Checker: The Results of Error Checking

>display desc

Comma with Adverbials:
   After running a mile   consider: After running a mile,

After running a mile he seemed tired.

Comma Use
To make your sentence easier to read or to signal a pause, consider using a comma to set off words or phrases (especially introductory words or phrases).

- Instead of: Unfortunately it rained the day of the picnic.
- Consider: Unfortunately, it rained the day of the picnic.

- Instead of: Once he got home he began to calm down.
- Consider: Once he got home, he began to calm down.
The MS Word Grammar Checker:
Controlling the Checker’s Behaviour

![Grammar Settings dialog box]

- Writing style: Formal
- Grammar and style options:
  - Capitalization
  - Commonly confused words
  - Hyphenated and split words
  - Misused words
  - Negation
  - Numbers
  - Passive sentences
  - Phrases
- Require:
  - Comma before last list item: don’t check
  - Punctuation with quotes: don’t check
  - Spaces between sentences: don’t check

SSLST 2011
Key Ideas

- A metric for ranking alternative parses [Heidorn 1982]
- Relaxation for parsing errorred sentences [Heidorn et al 1982]
- A heuristic fitted parsing technique for sentences outside the grammar’s coverage [Jensen et al 1983]
Outline

• What is a Grammatical Error?
• Grammar Checking without Syntax
• IBM’s EPISTLE
• Grammar Checking Techniques
• Related Areas
• Commercial Packages
Constraint Relaxation: The Basic Idea

- When a sentence cannot be parsed, relax the grammar rules in some way so that it can be parsed.
- The particular constraints that are relaxed indicate what the nature of the grammatical error is.
- First explored in the context of robust parsing by Weischedel and Black [1980]
Constraint Relaxation: Handling Constraint Violation Errors

- **Subject-verb number agreement**
  * John and Mary runs

- **Premodifier-noun number agreement**
  * This dogs runs

- **Subject-complement number agreement**
  * There is five dogs here

- **Wrong pronoun case**
  * He and me ran to the door

- **Wrong indefinite article**
  * A apple and a rotten old pear.
Constraint Relaxation: Handling Constraint Violation Errors

- A constraint in an EPISTLE rule:

\[
\text{NP VP (NUMB.AGREE.NUMB(NP)) } \rightarrow \text{VP(SUBJECT = NP)}
\]

- The same constraint in PATR-II:

\[
\begin{align*}
X_0 & \rightarrow X_1 X_2 \\
\langle X_0 \text{ cat} \rangle & = \text{VP} \\
\langle X_1 \text{ cat} \rangle & = \text{NP} \\
\langle X_2 \text{ cat} \rangle & = \text{VP} \\
\langle X_0 \text{ subject} \rangle & = X_1 \\
\langle X_1 \text{ num} \rangle & = \langle X_2 \text{ num} \rangle
\end{align*}
\]
Constraint Relaxation [Douglas and Dale 1992]: Relaxation Packages

\[ X_0 \rightarrow X_1 X_2 \]

\begin{align*}
1 \langle X_0 \text{ cat} \rangle & = \text{NP} \\
2 \langle X_1 \text{ cat} \rangle & = \text{Det} \\
3 \langle X_2 \text{ cat} \rangle & = \text{N} \\
4 \langle X_1 \text{ agr precedes} \rangle & = \langle X_2 \text{ agr begins} \rangle \\
5 \langle X_1 \text{ agr num} \rangle & = \langle X_2 \text{ agr num} \rangle \\
6 \langle X_0 \text{ agr num} \rangle & = \langle X_2 \text{ agr num} \rangle
\end{align*}

Relaxation level 0:
- necessary constraints = \{1,2,3,4,5,6\}
- optional constraints = \{

Relaxation level 1:
- necessary constraints: \{1,2,3\}
- relaxation packages:
  - (a) \{5, 6\}: Prenominal noun number disagreement
  - (b) \{4\}: *a/an error
Constraint Relaxation

• Advantages:
  – provides a precise and systematic way of specifying the relationship between errorful and ‘correct’ forms, making it easier to generate suggestions for corrections

• Disadvantages:
  – Requires significant amounts of hand-crafted linguistic knowledge
Mal-Rules

• Also known as error anticipation
• Mal-rules explicitly describe specific expected error forms
A Mal-Rule for Handling Omissions
[Schneider and McCoy 1998]

• Example:
  The boy happy

• Conventional rule:
  VP → V AdjP

• Malrule:
  VP[error +] → AdjP
Mal-Rules

• Advantage:
  – Specifically targets known problems
  – Allows easy identification of the nature of the error

• Disadvantages:
  – Requires error types to be catalogued in advance
  – Infeasible to anticipate every possible error

• Arguably mal-rules are just a notational variant of constraint relaxation approaches
Other Approaches

- Fitted parsing [Jensen et al 1983]
- Mixed bottom-up and top-down parsing [Mellish 1989]
- Minimum edit distance parsing [Lee et al 1995]
Outline

• What is a Grammatical Error?
• Grammar Checking without Syntax
• IBM’s EPISTLE
• Grammar Checking Techniques
  • Related Areas
• Commercial Packages
Robust Parsing

• The Goal:
  – Analyse extragrammatical input in order to extract some useful meaning
• No need to characterise and repair the error
• Processing of spoken language is a special case
Controlled Languages

• The Goal:
  – Ensure that a text conforms to a specific set of rules and conventions

• Examples:
  – ASD Simplified Technical English
  – Caterpillar Technical English
  – EasyEnglish
  – Attempto Controlled English

• See http://www.geocities.ws/controlledlanguage/
Outline

- What is a Grammatical Error?
- Grammar Checking without Syntax
- IBM’s EPISTLE
- Grammar Checking Techniques
- Related Areas
- Commercial Packages
Do Current Grammar Checkers Help?

- In real use, grammar checkers may have low recall and low precision
Kohut and Gorman [1995]: An Empirical Evaluation of Five Packages

<table>
<thead>
<tr>
<th>Package</th>
<th>Total # Errors</th>
<th>Real Errors Correctly Identified</th>
<th>Real Errors Incorrectly Identified</th>
<th>False Errors</th>
<th>False Errors/Total Detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerEdit</td>
<td>133</td>
<td>47%</td>
<td>12%</td>
<td>11%</td>
<td>16.13%</td>
</tr>
<tr>
<td>RightWriter</td>
<td>133</td>
<td>34%</td>
<td>8%</td>
<td>7%</td>
<td>13.85%</td>
</tr>
<tr>
<td>Grammatik</td>
<td>133</td>
<td>31%</td>
<td>6%</td>
<td>11%</td>
<td>23.44%</td>
</tr>
<tr>
<td>Editor</td>
<td>133</td>
<td>17%</td>
<td>3%</td>
<td>4%</td>
<td>16.13%</td>
</tr>
<tr>
<td>CorrectGrammar</td>
<td>133</td>
<td>15%</td>
<td>5%</td>
<td>10%</td>
<td>32.5%</td>
</tr>
</tbody>
</table>
## Kohut and Gorman [1995]: An Empirical Evaluation of Five Packages

### Mechanical Errors

<table>
<thead>
<tr>
<th>Errrors Found by Authors</th>
<th>PowerEdit</th>
<th>RightWriter</th>
<th>Grammatik</th>
<th>Editor</th>
<th>Percentage Correct Grammar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Punctuation</td>
<td>29</td>
<td>13 (45%)</td>
<td>9 (31%)</td>
<td>5 (17%)</td>
<td>5 (17%)</td>
</tr>
<tr>
<td>Agreement</td>
<td>8</td>
<td>2 (25%)</td>
<td>2 (25%)</td>
<td>3 (38%)</td>
<td>1 (13%)</td>
</tr>
<tr>
<td>Capitalization</td>
<td>2</td>
<td>0 (0%)</td>
<td>0 (0%)</td>
<td>0 (0%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Verb form</td>
<td>3</td>
<td>1 (33%)</td>
<td>1 (33%)</td>
<td>3 (100%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Sentence structure</td>
<td>20</td>
<td>15 (75%)</td>
<td>10 (50%)</td>
<td>9 (45%)</td>
<td>2 (10%)</td>
</tr>
<tr>
<td>Total mechanical errors</td>
<td>62</td>
<td>31 (50%)</td>
<td>22 (35%)</td>
<td>20 (32%)</td>
<td>8 (13%)</td>
</tr>
</tbody>
</table>

### Style Errors

<table>
<thead>
<tr>
<th>Errors Found by Author</th>
<th>PowerEdit</th>
<th>RightWriter</th>
<th>Grammatik</th>
<th>Editor</th>
<th>Correct Grammar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passive voice</td>
<td>15</td>
<td>9 (60%)</td>
<td>7 (47%)</td>
<td>4 (27%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Complex sentences</td>
<td>3</td>
<td>3 (100%)</td>
<td>3 (100%)</td>
<td>2 (67%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Wrong word</td>
<td>21</td>
<td>3 (14%)</td>
<td>4 (19%)</td>
<td>7 (33%)</td>
<td>4 (19%)</td>
</tr>
<tr>
<td>Redundancy</td>
<td>5</td>
<td>1 (20%)</td>
<td>0 (0%)</td>
<td>0 (0%)</td>
<td>1 (20%)</td>
</tr>
<tr>
<td>Weak wording</td>
<td>18</td>
<td>13 (72%)</td>
<td>7 (39%)</td>
<td>5 (26%)</td>
<td>6 (33%)</td>
</tr>
<tr>
<td>Slang/colloquialisms</td>
<td>2</td>
<td>1 (50%)</td>
<td>1 (50%)</td>
<td>0 (0%)</td>
<td>1 (50%)</td>
</tr>
<tr>
<td>Sexist language</td>
<td>6</td>
<td>0 (0%)</td>
<td>1 (17%)</td>
<td>1 (17%)</td>
<td>2 (33%)</td>
</tr>
<tr>
<td>Negative wording</td>
<td>1</td>
<td>1 (100%)</td>
<td>0 (0%)</td>
<td>0 (0%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>Total style errors</td>
<td>71</td>
<td>31 (44%)</td>
<td>23 (32%)</td>
<td>19 (27%)</td>
<td>14 (20%)</td>
</tr>
</tbody>
</table>
Welcome to StyleWriter

What is StyleWriter?

How does StyleWriter work?

Clarity and Readability

StyleWriter’s Editing Features

StyleWriter’s Scoring

Our newest release, StyleWriter v4.0, is a blazing-fast 32-bit application that is compatible with WINDOWS 7 and MS-WORD 2010! StyleWriter runs on YOUR computer and NEVER needs.
Grammarly is an automated proofreader and your personal grammar coach. Check your writing for grammar, punctuation, style and enhance your texts.

Check Grammar Now!
Instant reports with no registration

4,683,745 documents improved

Features and Benefits

150+ Grammar Checks
Check your text for the proper use of advanced grammar rules. Get accurate error explanations and

Grammarly vs. Other Products

Powerful Toolset:
42% Web Browser Extension
World-Leading English Writing Software

- Perfect your English writing with the most advanced editing tools: Spelling, Grammar Checker, Punctuation, Style and Structure
- Full Text Translation & Multilingual Dictionary: Translate full text in a single click and get usage examples for every word.
- Learn from your mistakes with our error explanation tool: Check every aspect of your text and learn how to avoid the same mistakes in the future.

Get it Now! Click here
English Software

"Detect The Most Difficult To Spot Writing Mistakes And Start Writing Professional, Rich, And Accurate English"

Includes World's #1 Grammar / Punctuation / Spell Checker

Check, correct and enrich your writing with one simple mouse click.

See Demo

Get It Now

All Inclusive Pack 2011

Advanced Grammar Checker
- The most powerful grammar checker on the market.

Contextual Spell Checker
- Contextual spell checking and an up to date word pool.

Style Checker
- Improve your writing style, vocabulary, and sentence structure.

Writing Review
- Learn from your mistakes and advance your skills at one go.

Text Enrichment Engine
- Get relevant ads and advice when you need them.
The World's #1 Grammar & Spell checker

Free Download

- Proofreading, spelling and grammar check in any application
- Improve your English writing - learn from your mistakes
- Text to speech reader - improve your pronunciation

Compatible with:

Try it yourself
Type/paste your text for correction and Ginger It!
PaperRater: Free Online Grammar Checker, Proofreader, and More

- Grammar & Spelling Check
- Free Online Proofreading
- No Downloads

Use Now FREE!

Grammar Checking
Robust grammar checking allows you to find those pesky mistakes and correct them before turning in your paper.

Plagiarism Detection
Find out if your paper contains plagiarized text before your professor does.

Writing Suggestions
Our proofreading system alerts you to opportunities to improve your writing.
John and Mary is coming today.

But the male participant was given more attention because he was interested. This method of interviewing was observed while performing a task, while the researcher makes notes, and perhaps records the interview on tape.

The main reason for the experiment is the belief that Unix editor vi.

However, has tended to use existing commercial manuals as a basis --- and the reduction of their original size, and to alter their contents to approach more closely the problem of trying to learn a new system.

Their feedback pointed to problem areas and causes for misinterpretation, and suggestions of improvements offered by them.

In this way, it is anticipated that the issue of native users not really knowing what it is they need to know is dealt with. All mailing systems have capabilities of composing, sending and receiving messages. The feature checklist was easy to administer and complete by experienced users.

Semi-structured interviews were conducted with experienced users to find what their most common tasks, the tasks a new user would need to begin, and what errors would be most likely in the early stages.

It had approximately 13% of the pages of the commercial manual, it allowed 30% faster learning and more effective use of the email system overall, and significantly better performance on individual subtasks including recovery from error.

The conditions under which our subjects worked tended to minimize such problems – since we asked them to persevere, and in the end they would be able to get human help.
Some Sample Grammar Problems

John and Mary is coming today.

A blocks are red.

But the males in this study experienced significant difficulties in this area and this problem suggest that some more attention be paid to the phenomenon.

This method requires a user to think aloud a task, while the researchers makes notes, and perhaps records the session on audio or video tape. The main reported problems was the Unix.

Both Carroll's work and our own, however, have used existing commercial manuals as a basis --- and the question then is how to prune to a fraction of the text, and to alter their contents to approach more closely to the problems that users actually confront. Their feedback pointed to problem areas a interpretation, and suggestions of improvements offered by them.

In this way, it is anticipated that the issue of native users not really knowing what it is they need to know is dealt with. All mailing systems have capabilities of composing, sending and receiving messages. The feature checklist was easy to administer and complete by experienced users.

Semi-structured interviews were conducted with experienced users to find what their most common tasks, the tasks a new user would need to begin, and what errors would be most likely in the early stages.

It had approximately 13% of the pages of the commercial manual, it allowed 30% faster learning and more effective use of the email system overall, and significantly better performance on individual subtasks including recovery from error.

The conditions under which our subjects worked tended to minimize such problems – since we asked them to persevere, and in the end they would be able to get human help.
Some Sample Grammar Problems

to the problems that users actually confront when trying to learn a new system. Their feedback pointed to problem areas and causes for misinterpretation, and suggestions of improvements offered by them.

In this way, it is anticipated that the issue of native users not really knowing what it is they need to know is dealt with. All mailing systems have capabilities of composing, sending and receiving messages. The feature systems had mailing capabilities that allowed users to compose and send messages to others.

Semi-structured interviews were conducted with experienced users to find what their most common tasks, the tasks a new user would need to begin, and what errors would be most likely in the early stages. It had approximately 13% of the pages of the commercial manual, it allowed 30% faster learning and more effective use of the email system overall, and significantly better performance on individual subtasks, thus helping users avoid errors.

The conditions under which our subjects worked tended to minimize such problems – since they persevered, and in the end they would be able to get human help. The more active but ineffectual behaviour of the males may mean that they feel they must persevere in order to overcome their errors and are less worried or affected by the possibility of making mistakes. Novice users should, however, be able to voice thoughts and desires on any topic, through the use of subtasks, the manual is to be properly user-centred.
Grammarly found 18 critical writing issues and generated 23 vocabulary enhancement suggestions for your text.

Score: 55 of 100 (weak, needs revision)
Conclusions

• Grammar checking is hard even for humans
• Automated grammar checking is a very unsolved problem
• Grammar checking is not necessarily distinct from spelling checking and style checking
• Many of the problems in real texts are more complex than straightforward textbook grammar errors, and often co-occur with other errors
• There’s lots to be done!