Ling/CSE 472: Introduction to Computational Linguistics

4/11/17

Evaluation

Overview

- Why do evaluation?
- Basic design consideration
- Data for evaluation
- Metrics for evaluation
    - Precision and Recall
    - BLEU score
    - Parseval
- Comparisons
- Error analysis
- Persistent evaluation issues
- Reading questions

But first: Term projects

- http://courses.washington.edu/ling472/final_project.html

Overview

- Why do evaluation?
- Basic design consideration
- Data for evaluation
- Metrics for evaluation
    - Precision and Recall
    - BLEU score
    - Parseval
- Comparisons
- Error analysis
- Persistent evaluation issues
- Reading questions

Why Evaluation?

- Good evaluation is essential to NLP research:
  - Verifies performance of process
  - Provides feedback on system changes
  - An essential part of the development process
  - Necessary for system comparisons
  - Provides information to potential users (and funders)

Ingredients

- Gold standard (“ground truth”)
- Evaluation metric: What you’ll count
- Baseline or baselines: What you’ll compare against
- Upper bound (optional)

Design considerations

- What system component is being evaluated? ex:
  - Parser
  - Language model
  - POS tagger
- What is the application? ex:
  - automated email response
  - travel dialogue system
  - document retrieval

Design considerations

- What are the evaluation criteria?
  - Accuracy
  - Coverage
  - Speed
  - Efficiency
  - Compatibility
  - Modifiability
  - Ease of use
  - Cost
  - …

Design considerations

- What is the goal of the evaluation?
    - Validation: Does the system do what you meant it to do?
    - Regression testing: Do recent changes improve performance, and/or lose any coverage?
    - Intrinsic evaluation: How well does it perform the specific task?
    - Extrinsic evaluation: How does it impact overall system performance?
    - Hypothesis testing: Can X information be used to aid in Y task?

Design considerations

- What resources are available?
  - Annotated corpora (e.g., Treebanks, aligned corpora)
  - Specialized corpora from application domain
  - Dictionaries and lexicons (e.g., pronunciation dictionaries, WordNet)
  - Test suites
    - Systematic collections of acceptable and unacceptable examples of specific phenomena
    - Generally hand built for each system and evaluation
    - Efforts to create shared resources, e.g. TSNLP (English, French, German)
- Are there standard corpora or evaluation metrics for the task?

Data for evaluation

- Separate test data from training and development data
- Use standard data sets where possible, to facilitate replication of results and inter-system comparison
  - Data often the result of challenges or shared tasks sponsored by NIST or various workshops
  - Data often distributed through LDC or ELRA
- Where there is no standard, clearly define the data and make it available to others

Handling data: Machine learning paradigm

- Divide data into training, development and test sets:
    - Training: Original input to stochastic model
    - Development: “Pretest” for tuning parameters (to avoid over-fitting on training data)
    - Test: Held-out data to measure generalizability of the system
- Dev and test data are always annotated (“gold standard”)
- Training data may be annotated (supervised learning) or not

Handling data: Knowledge engineering/rule-based paradigm

- "Training" data is examined by developer for rule development
- Training data is also used for regression testing
    - Does the current system analyze the same items as the previous one did?
    - Does the current system assign the same analyses as the previous one did?
- Test data is ideally unseen by both the system and the developer

Handling data: Knowledge engineering/rule-based paradigm

- Dealing with out-of-vocabulary words:
    - Measure overall performance anyway
    - Select only test data with known vocabulary
        - Add lexical entries for unknown words and test remaining system
- Error analysis can be very informative

Evaluation metrics

- Quantifiable measures
- Human inspection may be best, but can be impractical
- Automated approximations are cheaper, and especially valuable during system development
- The best metrics are those aligned with the goals of the application
- Use standardized metrics where available
- If none are available, clearly define the metrics used and use more than one

Example Metric: Precision and Recall

- Originally developed (and named) for Information Retrieval as a metric for search effectiveness

- Extended to the evaluation of various NLP tasks, especially ones involving categorization/labeling

- Provides measures of how correct (precision) and how thorough (recall); these goals are usually in tension

Precision and Recall

- Precision:
  - Proportion of results of the system that were correct

$$P = \frac{\#\text{correct results}}{\#\text{results returned}}$$

- Recall:
  - Proportion of correct results that were returned by system

$$R = \frac{\#\text{correct results}}{\#\text{results in gold standard}}$$

F-measure (combination of P and R)

$$F = \frac{(\alpha + 1) \times P \times R}{\alpha P + R}$$

- Varying the constant α affects the weight of Precision vs. Recall; increasing α increases the weight of Recall in the measure

- If α =1, Precision and Recall are equally weighted:

$$F = \frac{2 \times P \times R}{P + R}$$

Precision and Recall: Questions

- Why do we need to measure both precision and recall?
- Why would precision and recall be in competition?
- What is an example of an application that favors high recall?
- What is an example of an application that favors high precision?

Example Metric: BLEU score

- Automatic evaluation metric for machine translation (MT) (Papineni et al, ACL 2002)

- Measures similarity between system output and reference translations (gold standard)

- Measures lexical choice (unigrams), fluency (n-grams), and something like syntax (n-grams)

- Weighted average of the number of n-gram overlaps with reference translations: Weighted geometric mean of unigram, bigram, trigram and 4-gram scores

BLEU score

- Useful for comparing MT systems and tracking systems over time
- No meaningful units; for comparison, data sets must be the same
- One of several automatic MT evaluation metrics useful for development feedback
- Oft criticized
- Best MT evaluations use human raters (fluency, adequacy, edit distance)

Example metric: Parseval

- Automatic metric for evaluating parse accuracy when an annotated corpus is available
- Compares parser output to reference parses (gold standard)
- Evaluates component pieces of a parse
- Does not require an exact match: gives credit for partially correct parses

Parseval measures

- Labeled precision:

$$\frac{\text{\# of correct constituents in candidate parse}}{\text{total \# of constituents in candidate parse}}$$

- Labeled recall:

$$\frac{\text{\# of correct constituents in candidate parse}}{\text{total \# of constituents in gold standard parse}}$$

  - Constituents defined by starting point, ending point, and non-terminal symbol of spanning node

- Cross brackets: average number of constituents where the phrase boundaries of the gold standard and the candidate parse overlap

  - Example overlap: ((A B) C) v. (A (B C))

Issues with Parseval

- Parseval is the standard metric. However:
- Flawed measure:
  - Not very discriminating -- can do quite well while ignoring lexical content altogether
  - Sensitive to different styles of phrase structure (does particularly well on the flat structure of the Penn Treebank)
  - Too lenient sometimes, too harsh at others
  - Single errors may be counted multiple times
- Relevant only for CFGs (Phrase Structure Grammars)
- Most important question is: How well does it correlate with task improvement? Not clear.

Comparison

- Baseline: What you must beat
- Competing systems: What you want to beat
- Upper Bound (ceiling): What you aspire to

- Any difference must be statistically significant to count
- When comparing components, the rest of the system must be kept constant

Error analysis

- What types of errors does the system make?
- What are the likely causes of each error type?
- How could the system be improved?
    - Which changes would have the most impact?
- How do the errors affect larger system performance?
- Note difference between error analysis and debugging

Some persistent issues

- Development of test data and annotated corpora
- Development of generic and automated evaluation tools
- Creation of evaluation metrics
- Design of component evaluations that correlate well with application goals
- Development of multilingual data and evaluation techniques

Overview

- Why do evaluation?
- Basic design consideration
- Data for evaluation
- Metrics for evaluation
    - Precision and Recall
    - BLEU score
    - Parseval
- Comparisons
- Error analysis
- Persistent evaluation issues
- Reading questions

Reading questions

- How do we calculate P/R for term projects?

- What exactly is the difference between 'accuracy' and 'coverage' vs 'precision' and 'recall'?

- F-measure: What informs beta? I understand that the type of task determines the

- value, but what is a reasonable range? Why?

- What is the point of F-measure?

Reading questions

- What is the difference between training data, development data, and devtest data (pg. 277)?

- How do rule-based/non-statistical methods deal with over-fitting?

Reading questions

- Why do we need so many different kinds of evaluation metric? Is it perhaps system dependent? Varying more than just the categories given in each of the sub sections of section 3?

- How are intrinsic and extrinsic evaluations different from summative and formative evaluations?

- How exactly do researchers establish a correlation between results of automatic and manual evaluations?

Reading questions

- Are there any examples of machines trained against data generated by other machines, rather than humans? Are there areas of NLP which have surpassed humans in terms of accuracy AND precision?

- In this section, it mentions that it may be preferable to return no answer instead of potentially returning a wrong one. How can we measure the effects of an answer being wrong early in the pipeline, and to the degree in which it affects the final answer?

- The text mentioned ordinal scales, but didn't give any examples. Is this just because there aren't really any applications for this in NLP? It definitely seems like it would be less useful than interval and ratio scales.

Reading questions

- Could we possibly expand on the BLEU and ROUGE metrics, and what is the reason why they come with some controversy?
    - - bilingual evaluation understudy
    - - Recall-Oriented Understudy for Gisting Evaluation
- BLEU: Modified precision of n-grams compared to set of reference translations
- How do precision and recall relate to BLEU?

Reading questions

- What are similarities and differences between the word error rate and the translation error rate calculations?

- To me it seems that TER (section 3.4) would be rather arbitrary. What qualifies as an "exact" translation? What if the computer output is awkward but technically grammatically correct? Or if the output contains a word that that is similar to the desired meaning, but a slightly better word exists? Couldn't someone exploit the system by being more lenient on what counts as a good translation?

  - http://www.cs.umd.edu/~snover/tercom/

Reading questions

- Resnik and Lin talk about evaluation based on comparing an output probability distribution to a ground truth distribution. In what sort of applications would we have a "ground truth" distribution? It seems like if the ground truth distribution is flawed in any way, the evaluator would penalize the model. This would also be an issue when comparing other manually tagged data, but that seems less prone to error than generating a "true" possibility distribution.

Reading questions

- How is equation for cross-entropy derived? How does it relate to perplexity?

$$(6) \quad H = -\frac{1}{N}\sum_{i=1}^{N}\log_2 p_{\text{tri}}(w_i|w_{i-2}w_{i-1})$$

- '...perplexity is measuring the extent to which the model ptri correctly reduces ambiguity, on average, when predicting the next word in T given its prior context. To put this another way, on average we are ‘k-ways perplexed’ about what the next word will be,...' (p. 287) Does this mean that perplexity/cross-entropy is related to what the model is doing right or wrong? Also is this related to reducing the need to rely on human-evaluated systems?

- If one were to use an accuracy metric to evaluate a Word-Sense Disambiguation system, what would one have to do? If one then wanted to use a cross entropy evaluation, what would have to be changed?