

Ling/CSE 472: Introduction to Computational Linguistics

4/20/15

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

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 accuracy (precision) and coverage (recall)

Precision and Recall

- Precision (\approx accuracy):
 - Proportion of results of the system that were correct

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

- Recall (\approx coverage):
 - 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 $\alpha = 1$, 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 (ngrams), 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:

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

- Labeled recall:

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

- 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 standards
- 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

- I would like to see a couple more examples of the differences between training, development, development-test and test data. My understanding is that development data is a subset of training data that is used to improve performance after initially constructing the system using training data, while development-test data is a subset of test (rather than training) data, used for the same purpose. And then test data is data that is used when the system is completed, perhaps as proof of its efficacy, but not to improve the system.
- The reading describes formative and summative evaluations of NLP systems very generally/broadly toward the beginning of the reading. It mentions that formative evaluations are often 'automatic', but didn't say any more about what that might mean. Are they referring to a type of integrated testing tool?

Reading questions

- I am confused by the distinction between component and end-to-end evaluations. The readings mentioned that end-to-end evaluation is based on the output of a real tagger while the component evaluation is based on 'gold standard' supplied by a human. Aren't these two the same?
- In systems where the overall effectiveness is good yet the component effectiveness is poor (such as CLIR systems having bad translation components yet being roughly as good as monolingual IR systems at retrieving information), is the system considered successful if the overall effectiveness reaches the wanted level, or is it considered a poor system because it fails at tests of component effectiveness? If it's considered successful based on overall effectiveness, is there any reason to do component testing, or is end-to-end testing enough?

Reading questions

- What exactly are N-gram, and are we going to learn how to use them to do NLP? Also what is Kullback–Leibler distance in the context of the text?

$$(7) \quad D(p||m) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{m(x)}$$

- The article says "The most straightforward evaluation paradigm in NLP is one in which each input produces a single output" and "that output is compared against single correct answer". However, in many conditions, multiple possible outputs can appear, and those outputs are compared to several answers. So will this make the evaluation less accurate?

Reading questions

- Looking at word sense disambiguation, is there any metric to decide when multiple meanings are intended? If multiple meanings are intended, for puns or double meanings or whatnot, would both meanings just appear relatively equally probable to the algorithm?
- I know that these get referenced multiple times, but what exactly are the BLEU metric for machine translation and the ROUGE metric for text summarization?
- How do automatic evaluation metrics like BLEU, TER, and METEOR compare to each other? Do they have different purposes or evaluate the same thing in different ways? Are they commonly used together to provide multiple evaluations of one system?

Reading questions

- Why has it been so difficult to create a translation component that is better than word-for-word translation? Not that I'm saying that this task should be easy, but this field has created some amazing programs that have handled tasks that seem much more difficult, so I'm wondering what makes translation different?
- Among several answers, is there a ranking system exist? Just like a search engine, when we input certain keyword and countless output exist, so if similar case happens to NLP, how should we measure all the answers?
- Google returns millions of results for a general search term. This is considered low precision, however, google ranks the results in order of relevance. Does ranking in order of relevance affect precision?

Reading questions

- As the reading lists repeatedly how challenging it is to get human testers to give feedback and evaluate, I wonder if there is any tool online to connect people who are willing to help with researchers who need feedback for this? I've seen and participated in machine translation evaluations online, but not in labeling word sense. Furthermore, if such a tool did exist, I wonder if it would be desirable; it seems only getting feedback and tags from a certain demographic (bored people on the internet) could potentially bias the results, as it is something that can be subjective.