Unsupervised Learning for Natural Language Processing

(Morning Session)

Keh-Yih Su and Jing-Shin Chang
{kysu,shin}@bdc.com.tw

(1999/12/10)

Behavior Design Corporation
No. 5, 2F, Industrial East Road IV , Science-Based Industrial Park
Hsinchu, Taiwan 30077, R.O.C.

Table of Contents

Table of Contents (cont.)

Part I: Introduction

- Knowledge Acquisition in NLP
  - Tasks in NLP, Knowledge Representation Form
- Statistical Parameter Learning
  - Parameter learning, and why statistical parameter learning
- What is Unsupervised Learning
  - Characteristics & differences with supervised learning
- When Should Unsupervised Learning Be Used
  - Problem characteristics and suitable situations for unsupervised learning
- Why Unsupervised Learning is Becoming Popular
  - environmental factors & shift of paradigm

Knowledge Acquisition in NLP (I)

- Tasks for Building NLP Systems
  - Knowledge Representation
    - How to organize and describe linguistic knowledge
  - Knowledge Control Strategies
    - How to use knowledge for
      - efficient analysis
      - ambiguity resolution
      - ill-formedness recovery
  - Knowledge Acquisition
    - How to systematically and cost-effectively set up knowledge bases, and
    - maintain knowledge base consistency
  - Knowledge Integration
    - How to jointly consider various knowledge sources effectively

Knowledge Acquisition in NLP (II)

- The task of Knowledge Acquisition is usually the bottleneck
  - Language usage is complex (not governed by any elegant model), and dynamic (changing with different groups, locations, and time)
  - Required knowledge is huge, messy and fine-grained
  - Inducing rules by human is usually very expensive, and time-consuming
  - Traditional rule-based approaches are very hard to ensure global improvement, even if it is possible
  - Seesaw phenomenon is generally observed
- Knowledge can be represented in different Forms
  - Knowledge can be represented either explicitly (such as rules) or implicitly (such as parameters).
    - Example 1: IF C_{i-1} is *Det*, then C_i cannot be a *Verb*
    - Example 2: $P(C_i = Verb \mid C_{i-1} = Det) = 0$

Knowledge Acquisition in NLP (III)

- The Task of Knowledge Acquisition is closely coupled with the Knowledge Representation Form
  - Change the Knowledge Representation Form also change the way to acquire knowledge

- Consider the Knowledge Representation Form from the Knowledge Acquisition point of view
  - What kind of knowledge is suitable for machine learning?
    - Simple, uniform, easily to be derived from those observable data
    - Large quantity (once the number is large, even a collection of simple units, must be self-learnable, can also appear with smart behavior; for example, neurons and IBM Deep Blue, etc.)
    - Parametric form is most suitable for machine learning
  - Learning abstract forms (e.g., model) has not demonstrated its success in machine learning yet

Knowledge Acquisition in NLP (IV)

- Integrated Approach is better for Knowledge Acquisition (also classified as hybrid-approaches by some researchers)
  - Human derives parametric language model which possesses a lot of parameters
  - Parameter values are acquired from machine learning
  - The most promising approach in the next century

Types of Machine Learning (according to types of knowledge acquired):

- Symbolic: learning symbol relationship
  - Including: patterns, grammars, rules, decision trees, frames, networks, etc.
  - Example: Grammar Inference, Transformation Tagging [Brill 1994]
  - Advantages:
    - flexible,
    - acquired knowledge is compact and easy to interpret,
    - easily fit in existing linguistic theories
  - Disadvantages:
    - relatively awkward in dealing with complex and irregular decision boundary,
    - usually unable to achieve the best performance
  - Suitable for handling compact and regular situations

Types of Machine Learning (cont.):

- Parametric: learning parameter values under known parametric forms
  - Including: Neural-Net (learning weighting coefficients), statistical Language Model (learning statistical parameters), etc.
  - Example: Statistical Tri-gram Tagging Model [Church 1988]
  - Advantages:
    - acquisition mechanism is uniform and simple,
    - quantitative measure can be provided,
    - adaptability is high (provide good parameterized systems that can be controlled easily through various feedback mechanisms),
    - capable to achieve the best performance
  - Disadvantages:
    - parameter size is large,
    - acquired knowledge is not intuitive
  - Suitable for handling complex and irregular situations

Why Parametric Learning (I)

- NLP requires fine-grained knowledge
  - Inherited characteristics from the given problem
    - Different classes just don't have clean and regular separation boundaries between them
  - A lot of local descriptions are required
  - Huge messy knowledge required simple control mechanism to manage
  - Parametric approach is the ideal candidate

- Unsupervised-learning is the preferred operating mode in many different situations
  - Supervised-Learning requires annotating the corpus which is not affordable in many cases
    - Most symbolic learning algorithms operate only under the supervised-mode (I.e., the features based on which rules are induced must be observable)
  - Unsupervised Learning is not easy to go with symbolic approach
    - Un-supervised learning requires an objective measure to tell it where to go
    - It is difficult for symbolic learning to provide such an objective measure

Why Parametric Learning (II)

- Further Performance Push requires Quantitative Knowledge
  - Refined models required quantitative information in almost every fields (e.g., F= ma)
  - Detailed study unveils non-deterministic phenomenon
    - Non-deterministic is kind of quantitative knowledge
  - Quantitative model can outperform qualitative model
    - Qualitative model is a special case of the corresponding quantitative model
  - Symbolic approaches are not the suitable ways to provide the required quantitative knowledge
    - Rules only make Go or No-Go decision (I.e., a Hard-Rejection approach)

Why Parametric Learning (III)

- System Performance is the ultimate goal
  - Computer Memory and raw power are no more the issues
    - As Moore's Law keeps going, we only care about the scarceness of human resources not that of computer
    - Computer resource required for parametric learning is no more a constrain
  - Human power required for operating the system greatly depends on the system performance (e.g., Machine Translation, OCR, telephone switching system, etc.)
    - That is why we care about the error reduction rate (advancing from 98% to 99% makes sense)
  - Parametric approaches are more promising for delivering better performance

Neural-Net (or Connectionist):

- Learning weighting coefficients associated with those connection-links between neurons (a black-box approach): a universal approximator
- Input is usually a fixed-dimension static pattern
- Advantages:
  - Provide a quick solution: extensive problem analysis is not required
  - The mechanism is simple and easy to understand: a weapon for everybody
  - Suitable for real-time applications (architecture for parallel processing is implied)
  - Directly minimizing the error rate

Neural-Net (cont.):

- Disadvantages:
  - Not easy to handle the feature whose dimensionality dynamically varies (e.g., number of terminal-symbols under a constitute)
  - Not easy to handle the candidate of hierarchical structure with varying depth (e.g., linguistic constitutes), or the process with flexible duration (e.g., speech syllable)
  - Learning process is inefficient: requires relatively large amount of training data, and convergence period
  - Generalization capability is usually poor when training data is not abundant
  - Without truly understanding the problem, further improvement is difficult

Statistical Language Model (I):

- Learn those statistical parameters implied by the model (a glass-box approach)

- Advantages:
  - Decisions based on Bayesian classifier has a direct link with minimum error rate performance: the most promising approach to deliver the best performance

  - Supported by well-established statistics theories: we know why and how to improve the performance by using a lot of existing techniques

Statistical Language Model (II):

- Advantages (cont.):
  - Problems can be easily decomposed into more manageable and simpler explicit sub-problem (by introducing more intermediate random variables and then conditioning on them; using the theorem of total probability and multiplication rule)
    - Example: Machine Translation [Su 1995]

$$
\begin{aligned}
&P(T_i|S_i) = \sum_{I_i} P(T_i, I_i|S_i) \\
&= \sum_{I_i} P(T_i, PT_t(i), NF1_t(i), NF2_t(i), NF2_s(i), NF1_s(i), PT_s(i)|S_i) \\
&\cong \sum_{I_i} \{\left[P(T_i|PT_t(i)) \times P(PT_t(i)|NF1_t(i)) \times P(NF1_t(i)|NF2_t(i))\right] \cdots (1) \\
&\times \left[P(NF2_t(i)|NF2_s(i))\right] \qquad \cdots (2) \\
&\times \left[P(NF2_s(i)|NF1_s(i)) \times P(NF1_s(i)|PT_s(i)) \times P(PT_s(i)|S_i)\right]\} \qquad \cdots (3)
\end{aligned}
$$

    - where: S: source sentence, T: target sentence, I: intermediate normal forms
    - $I_i = \{PT_t(i), NF1_t(i), NF2_t(i), NF2_s(i), NF1_s(i), PT_s(i)\}$, in which
    - PT: parse tree, NF1: normalized syntax tree, NF2: normalized semantic tree
    - (1) = generation score (2) = transfer score (3) = analysis score

Statistical Language Model (III):

- Advantages (cont.):
  - Provide direct and flexible control to support those hierarchical internal structures (i.e., intermediate forms)
  - Model is more extendable (respect to model complexity) and scalable (respect to dimensionality of feature space) with the advancing of problem understanding and modeling
- Disadvantages:
  - Require statistical knowledge and modeling capability
  - Require problem analysis stage
  - Require an additional discrimination learning stage to compensate criterion mismatch

Why Statistical Parametric Learning (I)

- More suitable for unsupervised learning
  - Unsupervised Learning other than clustering is difficult with Neural Net approach
- More suitable for NLP application
  - Linguistic constitutes are not fixed-dimension patterns. They have hierarchical structure with varying number of terminal nodes and depth
  - Non-deterministic nature of NLP requires multiple candidates to be generated in early stages
  - NLP is usually a multi-stage process: we need more direct control over those intermediate forms

Why Statistical Parametric Learning (II)

- More promising to deliver better performance
  - Better results have been reported
  - With the aid of modeling, statistical approach is promising in generating better performance given the same fixed amount of training data (more efficient in data utilization)
    - With respect to the inherited complexity in NLP, the amount of available training data is still too limited (not enough to support those brute force approaches)
    - Model forms various equivalent classes, thus dramatically reducing the number of parameters required
    - Learning with parametric approaches (in statistics term) is more efficient than that with non-parametric approaches (e.g., Gaussian and Binomial distributions versus histogram) in data utilization
    - Additional Problem Knowledge (or Domain Knowledge), acquired through analyzing problem, add extra strength as research goes on (e.g., speech recognition)

Why Statistical Parametric Learning (III)

- More efficient training process
  - Theoretically, every model (can be converted to a mapping function) is able to be implemented with a universal approximator; however, learning every transformation from scratch is inefficient (requires lots of data)
    - Some neural-net approaches add a pre-processor to include feature transformation for promoting data utilization efficiency (e.g., LPC and State-Segmentation in speech recognition); however, this approach deviates from its advantage of simplicity
  - Statistical approaches offer relatively fast convergence speed and requires less processing power
  - More efficient training process ensures fast testing turn around time, and thus accelerate R&D advancing pace
- Real-time requirement is not a serious constrain now
  - With the Moore's Law keeps going, it is possible to implement many applications in software now (e.g., speech recognition)
  - Whether the architecture is more suitable for hardware implementation is thus less concerned during decision making

Elements of Parameter Learning (I)

- Performance Criteria: error rate (E), precision (P), recall (R), and F-measure
  - E = number_of_incorrect_identification / total_number_of_instances
  - P = number_of_correct_identification / number_of_candidates_in_list
  - R = number_of_correct_identification / number_of_correct_instances
  - F-measure (=2PxR/(P+R))
- Observations & Features
  - what to use for solving the problem (e.g., ambiguity resolution)

Elements of Parameter Learning (II)

- Statistical Language Model: Probabilistic Form & Parameter Values
  - Probabilistic form characterizes the relationship among features to reflect problem characteristics and make computation feasible
  - Knowledge is implicitly implied by (or distributed in) those large number of parameters
- Parameters Estimation & Learning (Adjusting) Process
  - Obtain a specific set of model parameters that can maximize the desired performance criterion

Elements of Parameter Learning (III)

- Training Corpus: known instances used for learning
  - The information source to learn the desired knowledge
  - The amount of implied Information is related to the Corpus Size and the Degree of Annotation (if under the supervised learning mode)
- Performance Evaluation
  - Performance is a statistic measure based on a set of finite samples
    - Values obtained from different sets of data (e.g., training set and testing set) are usually different
    - Has estimation errors as other statistics do
  - Estimation variance (i.e., estimation accuracy) depends on the size of the sampling data

Elements of Parameter Learning (IV)

- Data Set Classification
  - Training Set
    - The data set used to obtain model parameters
    - Performance measured in the training set reflects the model capability to fit available training instances
  - Testing Set
    - A data set which is independently sampled other than the training set
    - It is mainly used to measure the true system performance in the real world, which also reflects the model capability to fit other instances in the real world
  - Cross-Validation Set
    - Another set of data which is independently sampled other than both the training set and the testing set
    - It is mainly used to help making design decision (e.g., model complexity adopted, etc.)

More on Performance Evaluation

- Why testing set
  - The performance measured in the training set is generally over-optimistic (which is called over-fitting, or over-tuning, phenomenon). 100% accuracy is possible if the number of parameter is greater than that is needed
    - Over-fitting usually occurs when the number of training data is not enough to support the model complexity adopted
    - Over-tuning happens during the adaptive learning process while we have too many adjustable parameters that we can afford
  - We need another independent data set to reflect the true performance when the customer deploys the system in the real world
  - To keep the testing set away from contamination, it is not allowed to see (or involve) the details in any design/training phase
    - You cannot use it to decide the suitable model complexity, dimensionality of the feature space, or when to stop during the adaptive learning process

More on Performance Evaluation (cont.)

- Why Cross-Validation Set
  - Bad training set performance normally implies methodological flaw, so we can immediately know that we must re-do the design
  - In contrary, good training set performance may result from:
    - A really good model (which also get good testing set performance)
    - An over-fitted model (which will give bad testing set performance)
    - A set of over-tuned parameters resulted from the adaptive learning process (which gives bad testing set performance too)
  - As the testing set is not allowed to be used to help making design decision, we cannot disambiguate the above situations
  - We need another independent set of data to provide a simulated testing test performance to help us making design decision

Performance Evaluation Methods

- Re-substitution Estimate:
  - Use the same set of samples to design and test a model (training set performance)
- Holdout Estimate:
  - Use two mutually exclusive sets of samples to design and test a model
  - Less data is left in the training set; thus it would result in a worse system
  - The true testing set performance would be deteriorated, although its value can be more accurately estimated
- Leave-one-out Estimate:
  - Use one sample for testing and the other samples for design; test the model in rotation for each single sample, then report the performance by averaging the obtained result
  - Retain the largest amount of training set data (thus have the best model) while provide the most accurate testing set performance measure
  - Very time-consuming, as it demands to repeat the design process N times

Performance Evaluation Methods (cont.)

- Rotation Estimate:
  - Use one subset of the samples for testing and the other subsets for design; test the model in rotation for each subset
  - Example: 10-fold rotation
    - 1. Divide all data (D) into 10 subsets of equal size: $D = D_1 \cup D_2 \cup \ldots \cup D_{10}$
    - 2. For Iteration I =1 to 10:
      - use D_i as testing set, $D-D_i$ as training set, and evaluate testing set error rate E_i
    - 3. Report the estimated error rate as: $(E_1+E_2+\ldots+E_{10})/10$
  - A compromise between achieving better true testing set performance and obtaining more accurate measure on that

Modes of Learning

Supervised Learning

- Learning from Annotated Examples
- Example: part-of-speech tagging
  1. Collect data: the (det) design (n/v) of (prep) computer (n) ...
  2. Human annotation with correct parts-of-speech: the (det) design (n) of (prep) computer (n) ...
  3. Estimate Language Parameters according to annotation: P(n|det)=90/123, P(adj|det)=33/123, P(prep|n) =63/250, …, P(n|prep)=61/97, …
  4. Estimate likelihood and Conduct predictions: P(…, det, n, prep, n, …) = … 90/123 x 63/250 x 61/97 x …
- Just an estimation process (no iteration), if no adaptive learning process is adopted
- Advantage: capable to achieve better performance (as more information is carried by the annotation) given the same amount of training data
- Disadvantage: human annotation is usually time-consuming and expensive
  - Selective Sampling (I.e., select more effective new data for annotation) thus had been proposed to increase data collection efficiency

Modes of Learning (cont.)

- **Unsupervised Learning**
  - Learning with Un-annotated Examples
  - Example: part-of-speech tagging
    - do not have human annotation in Step 2
    - do not base on human annotation for estimating initial language parameters in Step 3
  - Advantage: human annotation is not required
  - Disadvantage: performance achieved usually is inferior to that of supervised learning
- **Bootstrapping:**
  - Learning with Un-annotated Training Data, however, start from an Annotated *Seed Corpus*
  - A compromise between the supervised learning and un-supervised learning
  - Provide most cost effective solution, if used appropriately

Decision factors for choosing appropriate Learning Mode (I)

- Problem Characteristics
  - Amount and Granularity of Knowledge required
    - Think about unsupervised learning when the answer is both “large” and “high”
    - Most NLP tasks require huge amount of linguistics knowledge
  - Inherent non-determinism in labeling (difficulty to annotate the corpus)
    - Choose un-supervised learning (or re-think the classification hierarchy) if the answer is “high”
    - Examples: part of speech tagging (62% of the words in Brown Corpus have only one tag) versus sense disambiguation (even experts have difficulty to assign appropriate tags)

Decision factors for choosing appropriate Learning Mode (II)

- Problem Characteristics (cont.)
  - Inherent Constraints (is the implied dependency strong?)
    - Choose un-supervised learning if the answer is “yes”
    - For example, bilingual corpus substantially eliminates impossible translations (or senses)
  - Inherent anchor points
    - Choose un-supervised learning if the answer is “abundant”
    - For example, un-ambiguous words in POS tagging task reduce possible tags of other enclosed words
- Resource Scarceness
  - Measured by: corpus size, amount of information required
  - Choose supervised learning if the corpus size is very limited

Decision factors for choosing appropriate Learning Mode (III)

- Cost for Preparing Learning Samples
  - Cost for Collecting Training Samples:
    - Have people do the data entry work (e.g., LEXIS-NEXIS)
    - From public resources (LCD, ROCLING)
    - From the web/news/bbs with automatic tools
  - Cost for Annotating Training Samples :
    - This is usually the bottleneck for supervised learning: requiring number of qualified persons for annotating the corpus and doing consistency check; besides, it also requires a long period of time for large scale projects
  - Choose unsupervised learning if you cannot afford the cost required
- Frequency of updating features
  - How often for changing dimensionality, values, or labels of the feature vector
  - Unsupervised-learning is prefer if the frequency is high

When to Use Unsupervised Learning (I)

- The task on hand requires huge amount of fine grained knowledge to achieve acceptable performance
  - Usually not affordable by providing supervised learning examples
- Inherent Constraints (or implied dependency) among the linguistic units are strong
  - Have a better chance to predict the best candidate through good language model
- Training data have enough explicit Inherent anchor points
  - Help to impose constrains on their neighbors
  - Make the task for resolving ambiguity easier

When to Use Unsupervised Learning (II)

- Good Language Model that can echo the human preference is available.
  - Have a better chance to learn language parameters that are capable to achieve satisfied performance
  - This criterion is actually required for both supervised-learning and un-supervised learning for achieving satisfied performance
- Uncertain in classification hierarchy:
  - re-annotation & re-training are required frequently; however, it is not affordable
- Uncertain in discriminative features to be adopted:
  - re-annotation & re-training are required frequently; supervision is costly

When to Use Unsupervised Learning (III)

- Mass amount of un-annotated data is available; however, annotation is not affordable or very difficult (highly confusing in assigning labels)
  - Information concentration (advantage own by the supervised-learning) is not essential
  - Constraints can be imposed to the corpus for helping to select more certain parts (e.g., selecting the boundary sentences of paragraphs for training the sentence segmentation model)
  - Implied information might be enough to cover the unobservable knowledge which would be, otherwise, directly provided in the supervised-learning case (from less amount of training data)

When to Use Unsupervised Learning (IV)

- Size of available resource keep increasing with time:
  - system parameters are to be updated frequently for incremental improvement
- **In Summary**: when the cost for supervised learning is high and unsupervised learning can achieve competitive performance, choose unsupervised learning

Why unsupervised learning is getting popular in NLP (I)

- The fact that NLP requires huge and find-grained knowledge is increasingly perceived.
- In general, better performance requires deeper analyses; however, the annotating task gets more and more difficult when the analysis gets deeper
  - The increase of inherent non-determinism make the task of assigning tags more difficult.
- Multi-lingual corpus is more available
  - Implicit constraints and implied annotations can reduce the degree of ambiguity

Why unsupervised learning is getting popular in NLP (II)

- Many public corpora only provide minimum degree of annotation (partially anchored); the cost for further annotation is beyond the reach of most researchers
  - The environment and supports for supervised learning is limited
- The size of on-line corpora increases rapidly in this Internet age
  - The degree of knowledge concentration is no more essential.
- The cost for possessing an un-annotated corpus diminish to almost nothing
  - Achieved through resource sharing (ROCLING, LDC etc.) or through acquiring from WWW; however, annotated corpora are still rare.

Why unsupervised learning is getting popular in NLP (III)

- Corpus-Based Statistic-Oriented (CBSO) approaches prevail in NLP community; however, it is a rapidly changing field
  - New model is tried in a fast pace
  - New features, classes are tested and refined rapidly
  - Very difficult to keep the associated annotation updated accordingly, if the supervised-learning approach is adopted