Text KM & Text Mining

An Introduction to Managing Knowledge in Unstructured Natural Language Documents

Dekai Wu

HKUST
Human Language Technology Center
Department of Computer Science and Engineering
University of Science and Technology
Hong Kong
http://www.cs.ust.hk/~dekai

Lecture Objectives

- Introduction to the concept of **Text KM** and **Text Mining** (TM)
  - How to exploit knowledge encoded in text form
  - How text mining is different from data mining
- Introduction to the various aspects of **Natural Language Processing** (NLP)
  - Introduction to the different tools and methods available for TM

Textual Knowledge Management

- **Text KM** oversees the storage, capturing and sharing of knowledge encoded in unstructured natural language documents
  - 80-90% of an organization's explicit knowledge resides in plain English (Chinese, Japanese, Spanish, …) documents – not in structured relational databases!
  - Case libraries are much more reasonably stored as natural language documents, than encoded into relational databases
  - Most knowledge encoded as text will never pass through explicit KM processes (eg, email)

Text Mining

- **Text Mining** analyzes unstructured natural language documents to extract targeted types of knowledge
  - Extracts knowledge that can then be inserted into databases, thereby facilitating structured data mining techniques
  - Provides a more natural user interface for entering knowledge, for both employees and developers
  - Reduces organizational resistance to maintaining KM artifacts
  - Reduces KE efforts

Natural Language Processing
Historical Perspective

- Timeline:
  - NLP is one of the oldest areas of Computer Science
  - One of science's grand challenges (Turing Test)
  - Paradigm shift – extremely rapid development (eg, the Google wars)
- NLP techniques reside at the interdisciplinary intersection of many independently evolved research directions
  - Knowledge-based AI
    - Computer science
    - Philosophy (mathematical logic, epistemology, ontology, semantics, pragmatics)
  - Statistical learning methods
    - Classical statistics (e.g., regression, curve fitting, …)
    - Statistical pattern recognition (speech recognition, OCR)
    - Machine learning (from symbolic AI) - induction of symbolic knowledge (rules, categories, frames, grammars, etc.)
    - Neural networks
    - Information retrieval
  - Linguistics & computational linguistics
  - Cognitive modeling & cognitive psychology
  - Neurobiology
  - etc.

No Magic Bullet: Language Dependence

- System performance is highly sensitive to what language the documents are in
  - Systems engineered for English are very poor at handling Arabic or Hindi text
- CJKV languages present particular difficulties
  - Encoding schemes
  - Non-alphabetic composition
  - Conceptual differences

No Magic Bullet: Domain Dependence

- System performance is highly sensitive to what domain the documents concern
  - The **language model** varies greatly for different domains
  - Systems engineered for one domain are poor at handling others
  - Systems engineered for broad domains are poor at handling narrow domains, and vice versa

Textbases: Infrastructure for Document Management & TM

- Dedicated document management systems
  - Pros:
    - Document hierarchies
    - Threads
    - Navigation
    - Metadata maintenance
    - Built-in search engines
  - Cons:
    - Limited extensibility, vendor lock-in
- File systems
  - Pros:
    - Ease of adoption
    - Flexibility
    - Backward compatibility
- Web / WebDAV distributed file systems
  - Pros:
    - Widespread use – the Web is the world's largest textbase!
    - Wide-range integration of web and file system – local PC to worldwide
    - Open standards – reduced vendor lock-in
  - Rapidly supplanting proprietary infrastructures

Tools

- Language identifiers
- Part-of-speech taggers (POS taggers)
- Morphological analyzers
- Parsers (shallow vs deep)
- Word sense disambiguators (WSD)
- Named entity recognizers (NER)
- Term extractors
- Semantic analyzers
- Indexers
- Generators
- Speech recognizers

Text Annotation

- Most tools need some way to represent the text along with the input and output features
- XML has emerged as the standard representation for all types of text annotation
- Ancestry:
  - Long history of markup languages
  - Simplified version of SGML (from NLP)
  - Text Encoding Initiative (from NLP)
  - Notational style popularized via HTML

Language identifiers

- First processing step is often to identify what language a document is in
  - Determines the appropriate subsequent processing stages
  - Many tools work with English only (or work poorly on other languages)
- Domain dependent
  - Identification accuracy is often sensitive to the domain
  - Incorrect identification leads to large errors further down the processing chain
  - Requires model training and tuning
- Issues:
  - Difficult encoding variants to identify (especially for CJKV text)
  - Mixed-language documents (very common in practice all over the world, especially in HK)

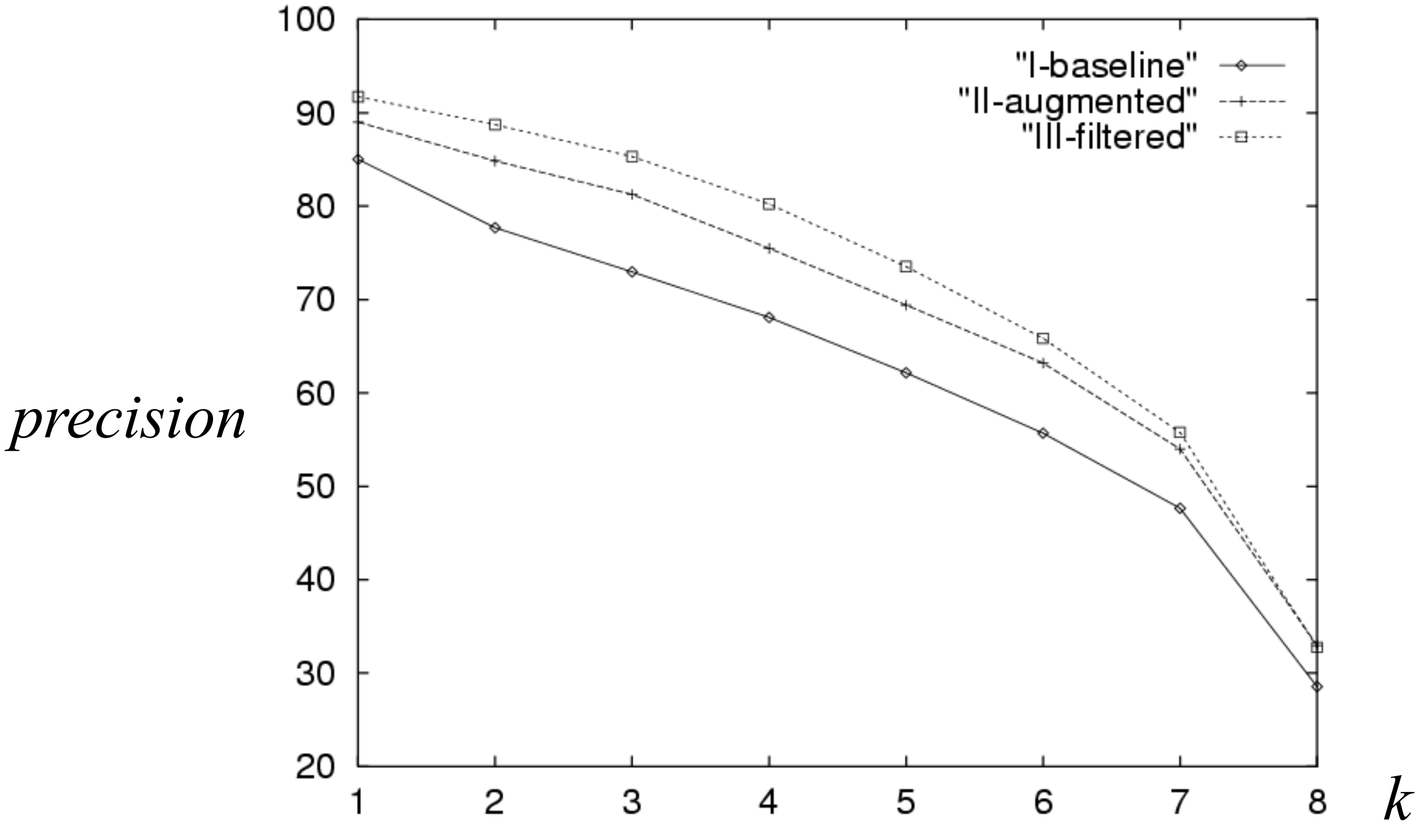
Morphological analyzers, word breakers, segmenters & tokenizers

- **Morphology** is the study of the structure and form of words
  - **Derivational**: undersimplification
    → under- + simplify + -cation
    → under- + (simple + -ify) + -cation
  - **Inflectional**: simplifies → simplify + -es (3rd person singular)

- **Stemmers** are extremely crude and simplified morphological analyzers
  - eg, simplifies → simplify

- Issues:
  - Most techniques don't work well for Chinese languages (eg, stemmers don't work at all)
  - Given a Chinese text, most people can't even agree exactly where the words (as opposed to the characters) are!

There's no *a priori* `correct' Chinese segmentation… Modifying the performance measure so that it rewards `fixed points' can impact scores heavily.



nk-blind precision comparisons for *n* = 8 judges (Wu & Fung 1994)

Part-of-speech taggers (POS taggers)

- **Parts of speech** are syntactic categories of words
  - eg, noun, verb, adjective, preposition, determiner, …
- **POS taggers** automatically annotate each word with the part-of-speech category

- All state-of-the-art systems employ statistical learning models

- Issues:
  - **Ambiguity**: though European languages can be tagged quite accurately, there are still errors – and even a single error can completely destroy subsequent processing of that entire sentence
  - **Segmentation**: how can Chinese words be tagged well, if it's not even clear where the word boundaries are?
  - **Lack of meaningful gold standard**: especially in Chinese, most verbs can also be used as nouns
  - **Effectiveness**: it's unclear whether POS taggers are needed, or whether parsers can do without them (below)

Parsers

- **Syntax** is the study of how words and phrases form sentences
- Consider Chomsky's classic example:
  - *Colorless green ideas sleep furiously.*
    vs.
    Sleep green furiously ideas colorless.
- **Parsers** analyze an input string and build a syntactic analysis of it
  - typically a **parse tree**
- A **shallow parser** does superficial analysis only
  - identifies major blocks (eg, base noun phrases)
  - does not attempt to build complete parse tree
- Issues:
  - **Lack of meaningful gold standard**: it is unclear what a "correct" parse means – even linguists disagree widely
  - Domain dependence: to be worthwhile, parsing must be used to improve some application, which is highly domain-dependent
  - Both issues are especially problematic for Chinese

Word sense disambiguators (WSD)

- **Semantics** is the study of meaning
- The first problem in semantic analysis is at the level of single words
- Issues:
  - **Lack of meaningful gold standard**: it is unclear what a "correct" word sense means – even linguists disagree widely
  - Chinese WSD is much more difficult than western languages

Word Sense Disambiguation (WSD)

Examples of test words

Target word	SMT	SMT + WSD
把握 bawo	0.1482	0.1484
包 bao	0.1891	0.1891
材料 cailiao	0.0863	0.0863
冲击 chongji	0.1396	0.1491
地方 difang	0.1233	0.1083
分子 fengzi	0.1404	0.1402
活动 huodong	0.1365	0.1465
老 lao	0.1153	0.1136
路 lu	0.1322	0.1208
起来 qilai	0.1104	0.1082
钱 qian	0.1948	0.1814
突出 tuchu	0.0975	0.0989
研究 yanjiu	0.1089	0.1089
运动 zhengdong	0.1267	0.1251
走 zhou	0.0825	0.0808

- State-of-the-art Chinese WSD model (Carpuat & Wu 2005)

Word Sense Disambiguation (WSD)
Examples of test words (cont)

- Example target word: 冲击 (*chongji*)
  - MT system alone inappropriately chooses "shock"

    "…against Japan for peace constitution shocks…"

  - WSD system forces using the more appropriate "impact"

    "…against Japan for peace constitution impact…"

Named entity recognizers (NER)

- Named entity recognition (NER) is a special case of WSD where the words to be disambiguated are:
  - proper names: persons, organizations, locations, …
  - temporal expressions: dates, times, days, …
  - quantities: ordinals, cardinals, percentages, currencies, …
- Configuration:
  - **Canonical forms** are useful
- Issues:
  - Accurate recognition of Chinese names is *extremely* difficult
  - Often hard to distinguish NER from general WSD

Named entity recognizers (NER)
A special case of WSD

The American envoy arrived Saturday evening and immediately asked to be shown the Real Madrid.

美国特使星期六晚上到达，立刻要求见识一下真正的马德里。

The Brazilian envoy arrived Saturday morning and was immediately brought to see Real Madrid.

巴西特使星期六早晨到达，立刻被带去观赏皇家马德里的比赛。

Named entity recognizers (NER)
A special case of WSD

The American envoy arrived Saturday evening and immediately asked to be shown the Real Madrid.

美国特使星期六晚上到达，立刻要求见识一下真正的马德里。

The Brazilian envoy arrived Saturday morning and was immediately brought to see Real Madrid.

巴西特使星期六早晨到达，立刻被带去观赏皇家马德里的比赛。

Semantic analyzers

- At a more general level, a **semantic analyzer** disambiguates an input text and constructs a semantic representation
  - may annotate the text with semantic relations, or roles and frames
  - may extract logical propositions, relations, or frames from the text
- Types of relations:
  - type-of (is-a)
  - part-whole (has-a)
  - frame roles
- May be used to assist ontology construction
- Issues:
  - **Lack of meaningful gold standard**: it is unclear what a "correct" semantic analysis means – even more than for syntactic analysis
  - **Accuracy and coverage**: While recent performance improvements are seen, semantic analysis is still extremely difficult

Semantic analyzers

- Semantics interacts heavily with syntax.
- Consider these two sentences:
  - *Time flies like an arrow.*
    vs.
    Fruit flies like a banana.
- Notice how the syntactic analysis differs, but the only way the parser could know this is via semantic analysis:
  - $(\text{Time}_{N}\ (\text{flies}_{V}\ (\text{like}_{Prep}\ (\text{an}_{Det}\ \text{arrow}_{N})_{NP}\ )_{PP}\ )_{VP}\ )_{S}$
  - $((\text{Fruit}_{Adj}\ \text{flies}_{N})_{NP}\ (\text{like}_{V}\ (\text{a}_{Det}\ \text{banana}_{N})_{NP}\ )_{VP}\ )_{S}$

Term extractors

- A **term extractor** or **collocation extractor** analyzes a corpus (large collection of text) to produce a list of domain-specific terminology
- More often than not, terms consist of multi-word expressions
- At a more sophisticated level, **semantic concept extractors** may identify synonymous terms, and may even attempt to place them within a hierarchical ontology
- A **multilingual term extractor** analyzes a multilingual corpus to identify terminology across languages (ie, translations)
- Issues:
  - Human-in-the-loop approaches are most effective
  - English models perform poorly on Chinese text

Grammar inducers

- A grammar inducer analyzes a corpus to automatically learn the grammar of the language (which can then be used to parse and analyze new sentences)
- Effective in limited domains and applications
- The general problem is technically an extremely difficult problem

Indexers

- An **indexer** builds indexes that facilitate efficient runtime access and retrieval to very large collections of text
- Most of the tools just described are essential to provide features that can be used to index the collections reasonably effectively
- Issues:
  - Domain and language customization is extremely important
  - Google-style indexing provides high precision, but low recall (ie, good but very incomplete retrieval of relevant documents)

Generators

- A **generator** takes a formal semantic representation of some idea or set of ideas, and generates one or more natural language sentences to communicate it to a human user
- Generation is the inverse of parsing
- Issues:
  - Much easier than parsing, but heavily constrained by what semantic representation is being used …
  - … which in turn is limited by the challenges of ontology construction

Speech recognizers

- **Automatic speech recognition** (ASR) or **speech-to-text** (STT) or, less accurately, **voice recognition**, like NLP, is another major subarea of human language technology
- ASR deals with spoken language rather than text
- Trend is toward integration of ASR and NLP components
- Issues:
  - Large-vocabulary, domain-independent speaker-independent, continuous, real-time ASR is technically extremely challenging
  - Many practical ASR technologies *do* exist

Applications

- Retrieval / search engines
- Categorization / routers
- Information extraction (IE)
- Question answering
- Summarization
- Translation
- Speech understanding

Retrieval / search engines

- As mentioned earlier, Google-style indexing provides high precision, but low recall (ie, good but very incomplete retrieval of relevant documents)
- Domain and language customization is extremely important
- Increased accuracy requires improved semantic analysis

Categorization / routers

- A **text categorizer** analyzes a text and automatically classifies it into one of many classes
  - Some systems can automatically classify into a hierarchical ontology (ie, a class hierarchy)
- Applications:
  - Organization
  - Recommendation
  - Filtering (eg, spam)
- Categories
  - Use a predefined ontology or thesaurus
  - Use clustering technologies to automatically produce an ontology
- Training
  - Highly domain-dependent; using out-of-the-box is unlikely to yield good results
- Number of categories identified per document
  - Multiple categories is usually better for robusness
- Topic weighting schemes
  - Intra-document ranking
  - Inter-document ranking

Information extraction (IE)

- An **information extractor** performs semantic analysis on a text or a corpus, so as to automatically extract targeted types of facts or proposition
- Example:
  - Input: "… *the acquisition of Compaq several years ago by HP resulted in long-term management changes…"*
  - Output: `merged(HewlettPackard, Compaq)`
- Applications:
  - Competitive intelligence gathering
  - Ontology construction
  - Search engine accuracy enhancement
  - Populating databases for data mining
  - …

Summarization

- A **summarizer** compresses a long document into a short text passage allowing users to quickly grasp the essence of the document
- Search engines often just use **KWIC summaries** (keyword-in-context summaries)
- More sophisticated **extractive summarization** extracts a few passages or sentences to serve as the summary
- Configuration:
  - Number of sentences to extract
  - Percentage of total text to extract
- Issues:
  - True summarization is extremely difficult as it requires deep comprehension of the text
  - Extractive summarization can be highly effective for quickly scanning large amounts of material

Translation

- **Machine translation** (MT) is one of the oldest subareas of CS and NLP
- Current trend is to incorporate **bilingual parsing** methods into **tree-based statistical MT** models (Wu 1995, …, 2008)
  - Addresses weaknesses in both traditional symbolic rule-based MT models and recent "dumb" statistical MT models
- Applications:
  - Text KM in multilingual workforces
  - Competitive intelligence gathering
  - Cross-lingual search, retrieval, categorization & recommendation
  - …
- Issues:
  - MT for Chinese is very different from western languages
  - Fully automatic high quality translation is extremely difficult as it requires deep comprehension of the text
  - Current MT can be highly effective for quickly scanning large amounts of material in a foreign language

Text KM & Text Mining
An Introduction to Managing Knowledge in Unstructured Natural Language Documents

Dekai Wu

HKUST
Human Language Technology Center
Department of Computer Science and Engineering
University of Science and Technology
Hong Kong
http://www.cs.ust.hk/~dekai