Universität Konstanz

MLP
Q&A – Summing Up

Miriam Butt, University of Konstanz

January 2015

The Course

Looking Back

- We began the course with a look at IBM's Watson.

- We also looked at PARC's Asker demo of a Q&A system.

- We have now understood most of the tasks that go into developing such systems.
  - Tokenization
  - POS Tagging
  - Morphological Analysis (Finite-State, Porter Stemmers)
  - Syntax
  - Semantics (Formal and Lexical)
  - Discourse Processing
  - Generation (Deep and "Canned" Text)

The Course

Looking Back

- The course has only been able to provide a rough overview of the
  - tasks
  - challenges
  - results/state-of-the-art

- We have also looked at Machine Translation (MT)
- Both MT and Q&A are highly complex and in some sense represent "ultimate" goals in Natural Language Processing (NLP)
- The most successful MT systems today use huge Translation Memories and statistical methods (very little linguistic knowledge).
- There are no truly successful Q&A systems – IBM's Watson is the best
  - but very domain specific
  - current deployment with North Face not very impressive (real life scenario)

Q&A Systems

IBM Watson

- IBM refers to this system as Deep Q&A system
- Components/Strategy
  - massively parallel probabilistic evidence-based architecture
  - incorporates all strategies from NLP
    - shallow approaches to parsing
    - deep approaches to parsing
  - heuristics/strategies for determining when to use which
  - sophisticated **information retrieval**
  - answer generation ("canned" text)

Information Retrieval

- Storage and Retrieval of all kinds of media.
- Main application so far is with *text documents* (also known as **Data Mining**).
- But work on pictures/videos is increasing.
- Text-based Information Retrieval:
  - **Document**: indexed unit of text indexed (e.g, a Webpage)
  - **Collection**: set of documents (e.g, the WWW).
  - **Term**: lexical item in a collection (e.g., *bass*).
  - **Query**: users informational need expressed as a set of terms (e.g., *Where can I catch bass?*).

Information Retrieval

- **Level of Sophistication:**
  - No information beyond the word.
  - **Bag of Words** approach is common: *I see what I eat* and *I eat what I see* are treated as equivalent.

- **Other Necessary Tasks:**
  1. Document Categorization
  2. Document Clustering
  3. Text Segmentation
  4. Text Summarization

Document Categorization

Classify a Document: Figure out which of an existing class of documents a given document should be identified as.

Most Common Method: Supervised Machine Learning

Good For:

1) Routing, e.g, getting e-mails to the right person to answer them.
2) Filtering, e.g., spam mails
3) Identifying the Language/Type of a Document, e.g., to retrieve only those

Document Clustering

- **Discover a Cluster of Documents:**
  - Maximize within-cluster document similarity
  - Minimize between-cluster similarity.
- **Efficiency:**
  - Clustering Documents allows for more efficient overall information retrieval.
- **Cluster Hypothesis** (Jardine and van Rijsbergen 1971)**:**
  - Identifying clusters should allow for greater precision/recall.
  - But, no good empirical support so far.
  - (More interesting recent work seems to be coming out of a study of how **Networks** work: comparing the WWW and human networks).

Precision/Recall

These measures are used generally to test the performance of a system. In terms of information retrieval, one can calculate the following:

$$\textbf{Recall} = \frac{\text{\# of relevant documents returned}}{\text{total \# of relevant documents in collection}}$$

$$\textbf{Precision} = \frac{\text{\# of relevant documents returned}}{\text{\# of documents returned}}$$

Evaluation

More Generally: How can the performance of a system be evaluated?

Standard Methodology adopted in NLP from Information Retrieval:

- Precision
- Recall
- F-measure (combination of Precision/Recall)

Evaluation

- **Establishment of a Gold Standard:**
  - Get a reference corpus and use it as a “Gold Standard” (benchmark)
  - This Gold Standard is usually annotated manually for whatever application is being targeted (POS-tagging, parsing, semantic annotation).
  - See how well the system performs with respect to the Gold Standard.
- **Recall:** Measure how much relevant information the system has extracted (coverage).
- **Precision**: Measure how much of the information the system returned is correct (accuracy).

$$\text{Recall} = \frac{\text{\# of correct answers given by system}}{\text{total \# of possible correct answers in text}}$$

$$\text{Precision} = \frac{\text{\# of correct answers given by system}}{\text{\# of answers given by system}}$$

Evaluation: F-measure

- Precision and Recall stand in opposition to one another.
- As precision goes up, recall usually goes down (and vice versa).
- The **F-measure** combines the two values.

$$\text{F-measure} = \frac{(ß^2+1)PR}{ß^2 P+R}$$

- ß can be set according to the needs of the system.
  - When ß = 1, precision and recall are weighted equally.
  - When ß is > 1, precision is favored.
  - When ß is < 1, recall is favored.

Text Summarization

Produce a shorter summary version of an existing document.

Knowledge Based:

- Detailed syntactic/semantic analysis which produces a meaning representation of the text.
- This representation is then passed on to a **generator**, which produces a new piece of text summarizing the original, longer text (this is the ideal world).

Selection Based:

- word frequency and discourse structure heuristics are used to identify the "important" sentences.
- A predetermined number of such important sentences are pulled out and included in the summary document.

Ad Hoc Retrieval

Ad Hoc Retrieval:

- An unaided user poses a question to a retrieval system.
- The system returns a set of ordered and hopefully useful documents.
- There are several possible methods of achieving this.
- The one most popularly used is the **Vector Space Method.**

The Vector Space Model

- Documents and queries are represented as **vectors of features.**
- The value of the feature indicates the presence or absence of a term (this could also be a weighted value).

Document: $\vec{d_j} = (t_{1,j}, t_{2,j,}, t_{3,j,}, \ldots t_{N,j,})$

Query: $\vec{q_k} = (t_{1,k}, t_{2,k,}, t_{3,k,}, \ldots t_{N,k,})$

The Vector Space Model –An Example

Document1: This is Miriam Butt's Web Page.

Vector of Features: [1, 1, 1, 1, 0, 0]

Document2: This is Tracy King's Web Page.

Vector of Features: [0, 0, 1, 1, 1, 1]

Query: Miriam Butt

Vector of Features: [1, 1, 0, 0, 0, 0]

Comparison: Figure out the number of terms two vectors have in common (via a similarity metric, J&M p. 697, (20.7.3).

Calculating Similarity

In the previous example:

- vectors were compared by simply summing the number of terms they share
- function words such as *this* and *is* or *the* and *and* are generally left out because they are not useful similarity indicators, see notion of "stop list".
- Terms are given a **binary** value: either they are found, or they are not found.
- However, some terms tend to be more important than others, so it is generally better to assign **weighted** values instead.

Term Weighting:

- Term Frequency:
  - Simple check to see how frequent a given term is in a document.
  - The assumption is that a frequently occurring term will be more important.
- Inverse Document Frequency**:**
  - Check for a term across a collection of documents.
  - The fewer documents a term occurs in, the higher its weight (i.e, it is a very important term in the context of that document).

Vexed Morphology

In a simple, term by term treatment, the following words will all be treated as completely unrelated terms:

process, processing, processed

This is clearly not desirable. One possible quick fix: integrate a stemmer (such as the Porter stemmer) to preprocess terms.

Problem: Throw away “too much” information. Example, not being able to distinguish *stockings* (<u>stock</u>) from *stocks* (<u>stock</u>) can prove to be extremely embarassing.

Stop List

Stop List:

- List of functional high-frequency words which are eliminated from a document
- These generally include elements such as determiners, conjunctions, auxiliaries.
- For English and other well-resourced languages, stop lists have generally been provided by somebody (e.g., NLTK).
- But they are not without problems:
    - *To be or not to be* could end up being looked up simply under "not".

Summary

- Much more work needs to be done on NLP.
- Many solutions do not involve much linguistic knowledge.
- But growing realization that some kind of **hybrid approach** is best (like IBM Watson).

- Course: Overview of main issues/tasks in NLP.
- The future:
  - learn more
  - in detail
  - contribute!