Exploiting Internal and External Semantics for the Clustering of Short Texts Using World Knowledge

Xia Hu,[1,2] Nan Sun,[1] Chao Zhang,[1] Tat-Seng Chua[1]

[1]School of Computing
National University of Singapore

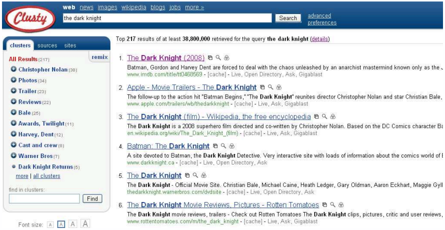
[2]School of Computer Science and Engineering
BeiHang University

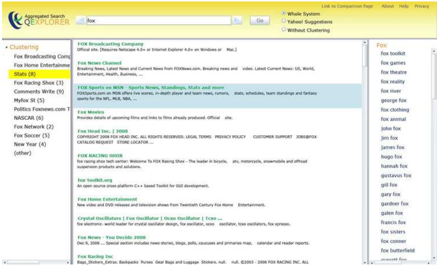
November 2, 2009

outline

Aggregated Search

- The form of browsing search results.





Short Texts

- Short texts, such as the snippets, product descriptions, QA passages and image captions etc., have played important roles in current Web and IR applications.
- Unlike standard texts with lots of words in length, short texts, which only consist of a few phrases or 2–3 sentences, especially present great challenges in clustering.
- Problems : "data sparseness" & "semantic gap".

Related Work

- Many methods have been proposed to improve the representation of standard text for clustering and classification, including “surface representation”[3,19] and “integrating world knowledge”[14].
- Several clustering techniques were employed to place the search engine snippets to their highly relevant topic-coherent groups[5,29].
- World knowledge bases have been found useful in improving the short text representation[1,23].

The General Framework

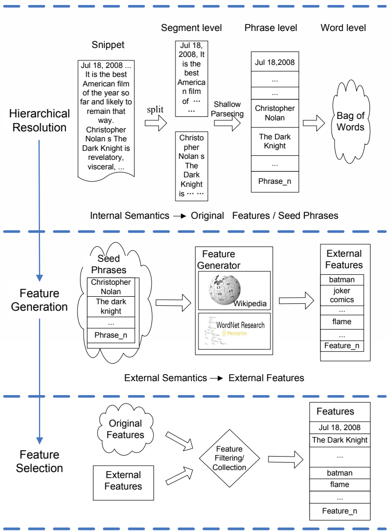


Fig: Framework for feature constructor

Hierarchical Resolution

"Jul 18, 2008 … It is the best American film of the year so far and likely to remain that way. Christopher Nolan's The Dark Knight is revelatory, visceral …"

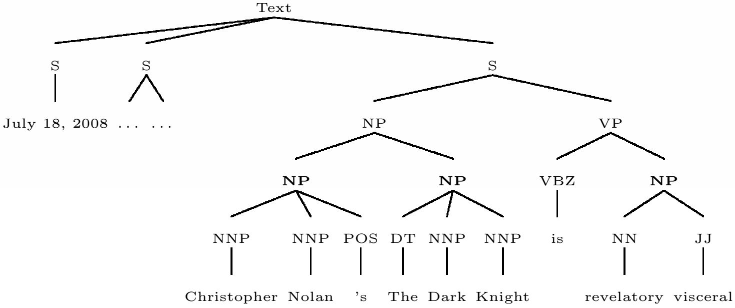


Fig: Syntax tree of the snippet

Original Feature Extraction

- Segment-level features.
- Phrase-level features.
  - *Sentence1* : [NP July 18 2008]
  - *Sentence2* : [NP It] [VP is] [NP the best American film] [PP of] [NP the year] [ADVP so far] and/CC [ADJP likely] [VP to remain] [NP that way]
  - *Sentence3* : [NP Christopher Nolan 's] [NP The Dark Knight] [VP is] [NP revelatory visceral]
- Word-level features.

Feature Generation

Two steps:

- the construction of basic features
  - seed phrases from internal semantics.
- the generation of external features.
  - external features from world knowledge bases.

Seed phrases selection (I)

- There are redundancies between *phrase level* features and *segment level* features.
- We propose to measure the semantic similarity between the two kinds of feature to eliminate information redundancy.
- For Wikipedia we download the XML corpus, remove xml tags and create a Solr index of all XML articles.

Seed phrases selection (II)

- Let P denotes a *segment level* feature, $P = \{p_1, p_2, \ldots, p_n\}$.
- We calculate the semantic similarity between p_i and $\{p_1, p_2, \ldots, p_n\}$ as $InfoScore(p_i)$.
- The p^* which has the largest similarity with other features in P will be removed as the redundant feature.

Seed phrases selection (III)

- Given two phrases p_i and p_j, the variants of three popular co-occurrence measures[6] are defined as below:

$$WikiDice(p_i, p_j) = \begin{cases} 0 & \text{if } f(p_i \mid p_j) = 0 \\ & \text{or } f(p_j \mid p_i) = 0 \\ \frac{f(p_i|p_j)+f(p_j|p_i)}{f(p_i)+f(p_j)} & \text{otherwise} \end{cases}, \quad (1)$$

where WikiDice is a variant of the Dice coefficient.

$$WikiJaccard(p_i, p_j) = \frac{min(f(p_i \mid p_j), f(p_j \mid p_i))}{f(p_i) + f(p_j) - max(f(p_i \mid p_j), f(p_j \mid p_i))}, \quad (2)$$

where WikiJaccard is a variant of the Jaccard coefficient.

Seed phrases selection (IV)

$$WikiOverlap(p_i, p_j) = \frac{min(f(p_i \mid p_j), f(p_j \mid p_i))}{min(f(p_i), f(p_j))}, \tag{3}$$

where WikiOverlap is a variant of the Overlap(Simpson) coefficient.

Linear normalization formula is defined below:

$$WD_{ij} = \frac{WikiDice_{ij} - min(WikiDice_k)}{max(WikiDice_k) - min(WikiDice_k)}, \tag{4}$$

A linear combination is then used to incorporate the three similarity measures into an overall semantic similarity between two phrases p_i and p_j, as follows:

$$WikiSem(p_i, p_j) = (1 - \alpha - \beta)WD_{ij} + \alpha WJ_{ij} + \beta WO_{ij}, \tag{5}$$

where α and β weight the importance of the three similarity measures.

Seed phrases selection (V)

For each *segment level* feature, we rank the information score defined in Equation 5 for its child node features at *phrase level*.

$$InfoScore(p_i) = \sum_{j=1, j \neq i}^{n} WikiSem(p_i, p_j). \tag{6}$$

Finally, we remove the *phrase level* feature p^*, which delegates the most information duplicate to the *segment level* feature P, and it is defined as:

$$p^* = \arg \max_{p_i \in \{p_1, p_2, \ldots, p_n\}} InfoScore(p_i). \tag{7}$$

Background Knowledge Bases

- Wikipedia, as background knowledge, has a wider knowledge coverage than WordNet and is regularly updated to reflect recent events.
- On the other hand, as the construction of WordNet follows theoretical model or corpus evidence, it contains rich lexical semantic knowledge.

Feature Generator

Algorithm 1: GenerateFeatures(S)

```
input  : a set S of seed phrases
output: external features EF

EF ← null
for seed phrase s ∈ S do
    if s.non-stop >1 then
        if s ∈ Segment level then
            s.Query ← SolrSyntax(s, OR)
        else
            s.Query ← SolrSyntax(s, AND)
        WikiPages ← Retrive(s.Query)
        EF ← EF + Analyze(WikiPages)
    else
        EF ← EF + WordNet.Synsets(s)
return EF
```

Fig: *External feature* generation scheme

Feature Selection (I)

Feature filtering for unstructured features:

- Remove features generated from too general *seed phrase* that returns a large number (more than 10,000) of articles from the index corpus.
- Transform features used for Wikipedia management or administration, e.g. "List of hotels"→"hotels", "List of twins"→"twins".
- Apply phrase sense stemming using Porter stemmer[24], e.g. "fictional books"→"fiction book".
- Remove features related to chronology, e.g. "year", "decade" and "centuries".

Feature Selection (II)

To avoid "curse of dimensionality":

- The number of *external features* we need to collect is determined by:

$$n_2 = \frac{n_1 \times \theta}{1 - \theta}. \tag{8}$$

- Select one external feature for each *seed phrase*.

$$f_i^* = \arg \max_{f_{ij} \in \{p_{i1}, p_{i2}, \ldots, p_{ik}\}} tf\_idf(f_{ij}). \tag{9}$$

- The top $n_2 - m$ features are extracted from the remaining *external features* based on their frequency.

Data Sets (I)

Reuters-21578 :

- We remove the texts which contain more than 50 words and filter those clusters with less than 5 texts or more than 500 texts.
- Thus it leaves 19 clusters comprising 879 texts. The number of texts in each cluster ranges from 6 (the cluster "income") to 438 (the cluster "acq").

Data Sets (II)

Web Dataset is built to simulate a real web application.

- As the users' interests are varied, we choose queries of different length according to the statistics of Google Trends during Nov. 26th 2007 – Nov. 25th 2008.

query length	One	Two	Three	more
count	4552	19762	6992	5290
percentage	12.4%	54.0%	19.1%	14.5%

- Ten hot queries are selected.

Tab: The selected hot queries in Web Dataset

NFL	Amazing Grace
Green Bay	Fox News Channel
60 Minutes	New York Giants
Total Eclipse	The Dark Knight
Black Friday	National Economic Council

Clustering Methods and Evaluation Criteria

K-means and *EM* are employed in this study. Six different text representation methods, as defined below:

- *BOW* (baseline 1) : Traditional "bag of words" model with the *tf-idf* weighting schema.
- *BOW+WN* (baseline 2) : *BOW* integrated with additional features from WordNet as presented in [14].
- *BOW+Wiki* (baseline 3) : *BOW* integrated with additional features from Wikipedia as presented in [1].
- *BOW+Know* (baseline 4) : *BOW* integrated with additional features from Wikipedia and WordNet as in baselines 2 and 3.
- *BOF* : The bag of *original features* extracted with the hierarchical view.
- *SemKnow* : Our proposed framework.

We evaluate performance of the methods using F_1 *measure* and *Average Accuracy*.

Performance Evaluation

Tab: Results using *k-means* algorithm

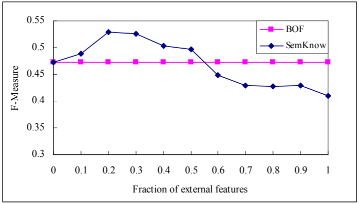
	Reuters-21578		Web Dataset	
	F_1measure (Impr)	AveAccuracy (Impr)	F_1measure (Impr)	AveAccuracy (Impr)
BOW	0.471 (N.A.)	0.550 (N.A.)	0.491 (N.A.)	0.563 (N.A.)
BOW + WN	0.473 (+0.43%)	0.552 (+0.26%)	0.530 (+8.01%)	0.576 (+2.30%)
BOW + Wiki	0.481 (+2.03%)	0.563 (+2.18%)	0.556 (+13.38%)	0.584 (+3.85%)
BOW + Know	0.489 (+3.75%)	0.566 (+2.86%)	0.558 (+13.79%)	0.583 (+3.70%)
BOF	0.473 (+0.33%)	0.551 (+0.19%)	0.520 (+5.95%)	0.570 (+1.24%)
SemKnow	**0.497 (+5.41%)**	**0.572 (+3.98%)**	**0.583(+18.81%)**	**0.586(+4.11%)**

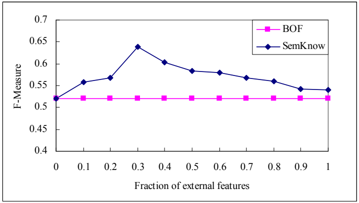
Tab: Results using *EM* algorithm

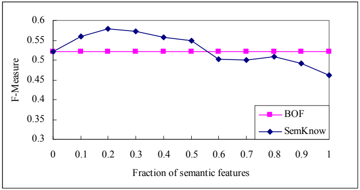
	Reuters-21578		Web Dataset	
	F_1measure (Impr)	AveAccuracy (Impr)	F_1measure (Impr)	AveAccuracy (Impr)
BOW	0.516 (N.A.)	0.579 (N.A.)	0.521 (N.A.)	0.608 (N.A.)
BOW + WN	0.525 (+1.72%)	0.585 (+0.99%)	0.540 (+3.59%)	0.626 (+3.02%)
BOW + Wiki	0.540 (+4.74%)	0.598 (+3.39%)	0.550 (+5.50%)	0.629 (+3.44%)
BOW + Know	0.542 (+5.13%)	0.607 (+4.54%)	0.556 (+6.74%)	0.635 (+4.41%)
BOF	0.520 (+0.82%)	0.594 (+2.63%)	0.536 (+2.73%)	0.624 (+2.55%)
SemKnow	**0.548 (+6.28%)**	**0.622 (+7.51%)**	**0.569 (+9.07%)**	**0.670 (+10.20%)**

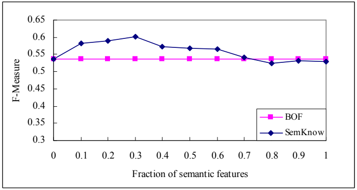
Effect of External Features

Impact of the parameter θ on Reuters and Web Dataset using $K - means$ and EM respectively.









Optimal Results

Tab: Optimal results using two algorithms

kmeans	**Reuters**	F_1meas(Impr)	AveAcc(Impr)
	BOW	0.471(N.A.)	0.550(N.A.)
	Optimal	**0.530(+12.35%)**	**0.604(+9.72%)**
	Webdata		
	BOW	0.491(N.A.)	0.563(N.A.)
	Optimal	**0.640(+30.39%)**	**0.607(+7.83%)**
EM	**Reuters**	F_1meas(Impr)	AveAcc(Impr)
	BOW	0.516(N.A.)	0.579(N.A.)
	Optimal	**0.578(+12.02%)**	**0.672(+15.40%)**
	Webdata		
	BOW	0.521(N.A.)	0.608(N.A.)
	Optimal	**0.602(+16.14%)**	**0.709(+16.56%)**

Conclusion

- In this study, we proposed a novel framework to augment the clustering accuracy of short texts by exploiting the internal and external semantics.
- The combination of internal and external semantics well tackled the problems of data sparseness and semantic gap in short texts.
- Empirical evaluations demonstrated that our framework significantly outperformed all the baselines including previously proposed knowledge-based short text clustering methods on two datasets.

Future Work

- As this work is for aggregated search, the efficiency of the whole framework should be optimized for real applications.
- Moreover, we will explore more tasks in NLP and information retrieval using the internal and external semantics generated by our proposed framework.

Thank you!