An Aspect Based Document Representation for Event Clustering

SA-OT accounts for pronoun resolution in child language

Wim De Smet, Marie-Francine Moens

Department of Computer Science
K.U.Leuven, Belgium

Abstract

We have studied several techniques for creating and comparing content representations of textual documents in the field of event detection. We define a document as a collection of aspects, i.e. disjoint components that reveal (latent) topics and/or extracted information such as named entities. As underlying models we consider the vector space model and probabilistic topic models based on Latent Dirichlet Allocation. We also investigate the value of dependencies between the aspects, which are reflected by importance factors. We apply and evaluate our techniques on event detection in Wikinews, where we cluster news stories that discuss the same event. We found that the split representations yield the best event detection results compared to the ground-truth event clusters. Our methods for aspect detection, for learning the importance factors of the aspects, and for event clustering are completely unsupervised.

1 Introduction

When processing news stories of several accounts of a certain happening, it is often relevant to determine whether two stories report on a same event. An event here is defined as a well-specified happening at a certain moment in time (a single day or a short period) which deals with a certain set of topics (e.g., a hurricane and inundations, an earthquake and lack of drinking water) and involves some named entities. Those entities are, for instance, the actors (such as the names of the leading persons or companies) and the location where the event occurred. News stories are typical examples. Broadcasted news can be segmented in different stories that each report on one event. Written news typically is recorded per story, where each story typically reports on one single event. However, different sources or the same source can produce several stories on the same event, which we might group as a preprocessing step for mining, summarizing of searching purposes.

In this article we focus on the clustering of textual news stories coming from different sources (we use the words "story" and "document" interchangeably). Any clustering depends on the quality of the distinction between the elements, and the quantitative representation hereof, i.e. the distance or dissimilarity function. Our main hypothesis is that comparing documents along different angles or aspects of their content enhances these distance computations. Based on the definition of a news event, we define these aspects to be the event's topics and entities.

Proceedings of the 19th Meeting of Computational Linguistics in the Netherlands
Edited by: Barbara Plank, Erik Tjong Kim Sang and Tim Van de Cruys.

We already have reliable named entity recognizers for common languages such as English that classify proper names into their semantic categories. Typical semantic categories are locations, persons and organizations. In addition, models for recognizing the topics in a text are well established. There is the long-standing vector space model (Salton 1989), and there are the newer probabilistic topic models, such as probabilistic Latent Semantic Analysis (Hofmann 1999) and Latent Dirichlet Allocation (LDA) (Blei et al. 2003).

The goals of this paper are to study and compare different methods of content comparison in text sources and to propose novel techniques of content splitting in order to quantify more accurately the similarities (and possibly differences) in content, thereby improving content-based clustering. A requirement we place on our methods is that they are completely unsupervised.

The remainder of this paper is organized as follows. Section 2 describes our methodology. Comparative tests and evaluations are presented in section 3. Section 4 discusses related work. In section 5, we present our conclusions.

2 Methodology

2.1 Document representations

Our first task is to create a document representation d_i. We consider several approaches. A document can be described as a term vector or a set of term vectors (vector space model), or probabilistic content models can be built from the document. For each approach we consider models that do not make a distinction between the words of the text (full text models) and models that split the documents into components or aspects based on semantic information. The aspects we consider for news event documents are the *entities* and the *topics*.

Named entities are entities in the real world that have unique names. Different types of named entities occurring often in news reports are for instance persons and organizations (the actors of an event), locations (where the event takes place) and timestamps. Named Entity Recognition (NER) detects and classifies the entities. We use the OpenNLP[1] package, which detects noun phrase chunks in the sentences that represent persons, locations and organizations.

The topics of a news event are the generally applicable subjects. For example, in a story about an earthquake, topics may be the earthquake itself, damage to houses, flooding, etc. With "generally applicable", we mean that every story on earthquakes might contain these words. We consider everything in a news story that is not a named entity as part of a topic.

Vector Space Model

In the vector space model (Salton 1989), a document is represented as a vector in a n-dimensional space: $d_i = [w^i{}_1, w^i{}_2, \ldots, w^i{}_n]$, where n is the number of used features. The features w_i commonly represent the terms of the vocabulary by

[1] http://opennlp.sourceforge.net

which the documents in the collection are indexed. Term weights might be binary indicating term presence or absence, or have a numerical value to indicate the importance of the terms in the document, for instance, weights are often computed by a $tf \times idf$ weighting scheme.

When representing our full text, we use one vector containing all terms in the document. To represent our different aspects, we use vectors that contain only the relevant part of the content. On one hand we have a topic vector which consists of all words that have not been classified as a named entity, a stopword, or have a low idf. On the other hand we have either an entity vector containing all named entities, or three entity vectors when we split them according to their semantic class (person, location, organization).

Probabilistic Model

In the example of an earthquake event, we mentioned that it may cover topics such as flooding, damage, etc. Probabilistic models define a mathematical basis for this idea. One can define a number of topics, each characterized by a probability distribution over words. An event can be seen as a mixture of these topics, where some topics are prominently and others only marginally present. As we want an unsupervised approach, we need a way to automatically define and detect these topics. For this purpose, Latent Dirichlet Allocation or LDA is used. LDA is a statistical model for document generation, presented in (Blei et al. 2003). The idea is that documents are created according to a random mixture of topics, sampled from a topic distribution. These topics generate a random set of words, sampled from each topics word distribution. LDA learns both kind of distributions in an unsupervised way, based on a training set of documents.

By learning corpus-independent parameters, we can infer topic distributions on new, unseen documents, that are compatible with the topic distributions of the training set. It has been shown in the literature that, if the training set is large and diverse enough, the topic-word distributions are stable. Typical values for the number of topics to be useful lie in the range [100, 300] for the English language.

The power of LDA lies in the natural modeling of synonymous and related words and of polysemous words. Another advantage is the possibility of inferring the topic distributions of new documents. In certain settings this inference is very useful. For instance, when dealing with a stream of news stories, new events are added continuously, making a frequent retraining of the system inconvenient.

When LDA is trained on the documents' full texts, the entities are part of the topic distributions. This has the undesirable property that entities that were not apparent in the training set (which, given the dynamic nature of news, occurs often) can not influence the topic inference of a new event. Therefore, we also train LDA on documents where the entities have been removed first. Due to the shared use of the term topic, both meaning a word-distribution in LDA as the content of a news story, confusion may arise. The context will help to disambiguate between the two.

Because named entities in news change dynamically (e.g. person, location and

organization names occur which never had been mentioned before), named entity models are difficult to learn from text corpora. Therefore, we chose a different probabilistic representation of entities. We create a probabilistic distribution, much in the same way as we would create a vector in the vector space model. Normalization (division by $\sum_j {d^j}_i$, not $||d_i||$), ensures the property of summation to 1. This applies both for the models based on all entities together, or separated by their class.

2.2 Dissimilarities between aspect representations

The similarity between two document vectors is computed as the cosine of the angle between the two normalized vectors, thus the dissimilarities between two documents d_i and d_j becomes

$$dis(d_i, d_j) = 1 - cos(\widehat{d_i, d_j}) = 1 - d_i \cdot d_j.$$

This dissimilarity can be computed considering the term vector of the documents containing all terms, or by considering the separated representations (e.g. named entities).

In case the documents are represented with a probabilistic content model, the probability distributions are compared with the symmetric Kullback-Leibler divergence of the n-dimensional probability distributions d_i and d_j, defined as

$$KL(d_i, d_j) = \frac{1}{2}\Big(\sum_{l=1}^{n} {d_i}^l \log(\frac{{d_i}^l}{{d_j}^l}) + \sum_{l=1}^{n} {d_j}^l \log(\frac{{d_j}^l}{{d_i}^l})\Big)$$

where ${d_i}^l$ is the probability of the l-th dimension of d_i. For entities, d_i is the term vector normalized by its sum, for LDA generated topics it is the topic distribution associated with the document.

We found that, when dealing with typical topic and/or entity distributions, the average KL divergence is dependent on the number of elements in the distributions. As most aspects will contain a different number of elements, this creates a scaling problem. We therefore normalized each divergence by dividing it by the maximum divergence for its dimensionality. Theoretically, the KL-divergence is unbounded. However, topic probabilities inferred by LDA using variational inference (Blei et al. 2003)[2] have a lower bound, equal for all topics that had no words associated with them in the document. When we know how many of these topics are apparent in each document, we can calculate an upper bound to the Kullback-Leibler divergence, by assuming the documents have no topics in common. For named entity distributions we use a default lower bound value, as they have not been calculated by LDA. Dividing the divergence by this upper bound yields a value between 0 and 1.

[2]Using Blei's implementation at http://www.cs.princeton.edu/blei/∼lda-c/

2.3 Combining Aspects

The content models above allow comparing different aspects of documents. To compare documents as a whole, we need to combine the dissimilarities between each of the aspects of the documents. Formally, for a document d_i we have defined the aspects of topic (${A^t}_{d_i}$), entities (${A^e}_{d^i}$), which can alternatively be split in persons (${A^p}_{d_i}$), locations (${A^l}_{d_i}$) and organizations (${A^o}_{d_i}$). For comparing aspects represented in the vector space model, we use the 1-complement of the cosine metric, as seen above. In a probabilistic setting, aspects are compared by computing the divergence of their probability distributions. Our normalized KL-divergence has also the properties of a dissimilarity function.

The obtained dissimilarities between different aspects can be combined in several ways to obtain a global document dissimilarity. We propose two ways of combining them:

$$\begin{aligned} \text{max:} \quad & max_k dis({A^k}_{d_i}, {A^k}_{d_j}), \quad k = 1 \rightarrow N \\ \text{average:} \quad & \tfrac{1}{N} \sum dis({A^k}_{d_i}, {A^k}_{d_j}), \quad k = 1 \rightarrow N \end{aligned}$$

where N is the number of aspects the document is split into: N = 2 for the topic-named entity split, N = 4 for the topic-person-location-organization split.

Each of these combination functions imposes different views of what is important when comparing documents. The max-function ensures that two documents are dissimilar when at least one of the aspects has dissimilar distributions: if two documents differ too much in one aspect, then it does not matter whether the other distribution is close or not. In an event setting, this translates into the following: if we detect different actors or locations, then we assume that we deal with different events, even when their topics are similar. Analogically, events with different topics that happen at the same location will be treated as different events.

The average-function is more tolerant towards differences. Even when covering the same event, different sources may stress different locations, interview different persons, etc. However, as named entity recognition is not yet perfect, it is possible (as we have encountered in our evaluations) that essential, shared entities are not recognized. This makes the named entity distributions divergence larger then it should be. Averaging with the topic distribution dissimilarity smooths these differences.

2.4 Clustering

The document dissimilarity $dis(d_i, d_j)$, which is a fused dissimilarity in case documents are represented with different aspects, is used in a clustering algorithm. We used a hierarchical agglomerative clustering with complete linkage, as it is mentioned in the literature as one of the best performing document clustering algorithms (Voorhees 1986). The hierarchical clustering algorithm does not require the number of clusters to be chosen a priori, a very important property in our dynamic environment. We can use a fitness-condition on the clustering to create a

natural, unsupervised stopping criterion. This natural clustering is the most logical extension of our unsupervised approach: the data provides the number of clusters itself. The clusterings fitness is calculated as follows.

For every document d_i in our corpus, we calculate its fitness in cluster C_i as the normalized difference between the distance of d_i to the second best cluster C_j, and the average distance of d_i to the other documents in C_i:

$$\begin{aligned} f(d_i) &= \frac{b(d_i) - a(d_i)}{\max\{a(d_i), b(d_i)\}} \\ \text{where} \quad a(d_i) &= \frac{1}{|C_i| - 1} \sum_{d_j \in C_i} dis(d_i, d_j) \\ \text{and} \quad b(d_i) &= \arg\min_{C_j} \frac{1}{|C_j|} \sum_{d_j \in C_j} dis(d_i, d_j) \end{aligned}$$

If C_i is a singleton cluster (containing only d_i), we assign $f(d_i)$ the default value 0. We search for the clustering that maximizes the average of f over all documents, over all possible stops in the hierarchy.

2.5 Importance Factors

Considering increasingly more aspects of a document is no guarantee of improving document similarity. By splitting the named entities into their semantic classes (as in our case into persons, locations and organizations), the individual classes might suffer a data sparseness problem. If no, or only a few entities of a class were present in the document, than that class' distribution divergence to other distributions is more sensitive to small differences. In some cases this is desirable: in case of two disaster reports which only mention the locations name, different names have to be able to discriminate the two events. On the other hand, if each report mentions a different person (for example in an interview), the documents would be discriminated based on irrelevant information. This notion is exactly what defines our aspects importance: the similarity of the distributions throughout different coverings of the same content, in our case the same event. The aspect's importance factor is the quantative representation hereof.

Our algorithm for learning the importance factors of topics, persons, locations and organizations starts from the output of the event clustering when using the topic-entity split. We assume that the output of this first step is sufficiently accurate so that we can bootstrap from it. Essentially, we wish to apply our technique to extract latent information from a training set, that we can then apply in a second step.

We have defined a document d_i as a collection of N different aspects, each having its own aspect distribution: ${A^k}_{d_i}$, for $k = 1 \rightarrow N$. We wish to associate with each document d_i a N-dimensional vector ${H}_{d_i}$, whose k^{th} dimension gives the relative importance of the k^{th} aspect. The elements of ${H}_{d_i}$ should summate to 1.

To learn the importance factors from a corpus C, for each document d_i in C we calculate H_{d_i} as follows:

---

for all $d_i \in C$ **do**
  Set ${H_{d_i}}^k = 0$ for each k in $1 \rightarrow N$
  for all $d_j \in C, d_j \neq d_i$ **do**
    Calculate the similarity of d_i and d_j using a similarity measure: $sim(d_i, d_j)$
    for all $kin1 \rightarrow N$ **do**
      ${H_{d_i}}^k += sim({A^k}_{d_i}, {A^k}_{d_j}) \times sim(d_i, d_j)$
    end for Normalize H_{d_i}
  end for
end for

$sim(d_i, d_j)$ is the similarity between two documents, defined as the 1 - complement of $dis(d_i, d_j)$.

Once we have trained on a corpus C, we can calculate the importance factors H_{d_n} of a new document d_n by taking the weighted average of the H_{d_i}'s of training documents d_i, weighted by their similarity to d_n.

The values are then used as weights for the combination functions:

$$\begin{aligned} \text{weighted max} &= \max_k \left(dis({A^k}_{d_i}, {A^k}_{d_j}) * \times {H^k}_{d_i} \right) \\ \text{weighted average} &= \sum_{k=1}^{N} dis({A^k}_{d_i}, {A^k}_{d_j}) \times {H^k}_{d_i} \end{aligned}$$

Note that these combination functions are asymmetric, as the importance factors are associated with the first of the two documents. Due to the clustering algorithm, the smallest of these gets chosen.

3 Evaluation

We will first give details on the datasets used in the evaluation of the event clustering. Then follows a short section on our clustering algorithms and cluster evaluation techniques. After that, we present results and discussion.

3.1 Datasets

For our evaluation, we used three different datasets: 1) TREC, 2) Reuters and 3) Wikinews, each for a different goal. The TREC dataset is used to train our LDA model. From Reuters we learn the importance factors for the different aspects, which we then evaluate on Wikinews. A more detailed description of each dataset is given here.

TREC The training set for the topic model needs to cover a wide range of topics, in order to have clean word distributions. From the Text Retrieval Conferences

TREC Vol. 5, we randomly selected over 30, 000 documents out of the LA Times corpus, reporting events from areas as different as the political, financial or scientific world, the world of media and entertainment, etc. After removal of stop words, low-idf words (2.0) and named entities. We ended up with a word-list of $50,000$ elements. We used Bleis LDA-utility[3], to create topic-distributions of size 100.

Reuters We used part of the Reuters corpus to train the importance actors as explained in section 2.5. This corpus consists of 10,223 documents from the LA Times newspaper, reporting on events from diverse news domains.

Wikinews To test our different techniques for event clustering, we need a corpus for which we know of every document which event it covers, and to which other documents it relates. We considered using the TDT4 corpus. The large number of documents (28, 500) is a positive point; however, only 160 separate events are annotated. This makes a realistic computation of precision and recall impossible. Therefore we created our own evaluationa corpus3 from Wikinews. On this news website, every reported event comes with several links to sources from different news-providers, thus providing a set of documents which cover the same event. We collected 1,000 documents in two runs, covering 327 separate events that happened between Jan. 1 and Jan. 24, 2007, and between Dec. 1 and Dec. 21, 2007. Each event is covered by an average of 3.05 documents, with the number of covering stories for each event ranging from 1 to 10.

3.2 Evaluation metrics

The evaluation of our clustering is done using the B-Cubed metric (Bagga and Baldwin 1998). Let C_i be the symbol for the cluster that document d_i gets clustered in, and M_i be its manual cluster (i.e. from the ground truth). The B-Cubed metric then calculates for each document its precision (how many of the other documents in its automatic cluster should be in it?) as $\frac{|C_i \cap M_i|}{|C_i|}$, and its recall (how many of the documents in its manual cluster are in its automatic cluster?) as $\frac{|C_i \cap M_i|}{|M_i|}$. The total clusterings precision and recall are taken as the average over all documents.

Our main remark on the B-Cubed metric is the fact that it rewards a singleton clustering (each document in its own cluster) with a precision of 100%, as no document is clustered together with an unrelated one. Of course, recall will be very low in that case. Therefore we present the F1 values, as these give a clear view on both precision and recall.

[3] http://www.cs.princeton.edu/ blei/lda-c/

Vector space	F-measure	# events	*Probabilistic*	F-measure	# events
Full text	76.8%	208	Full text	69.5%	119
Topic words	65.6%	162	Topics	59.5%	119
Entities	62.1%	173	Entities	66.1%	182
max	85.7%	271	*max*	72.7%	213
average	67.3%	164	*average*	72.7%	213

Table 1: F-measure and # events for the 2-way split vector space and probabilistic models

Vector space	F-measure	# events	*Probabilistic*	F-measure	# events
Full text	76.8%	208	Full text	69.5%	119
Topic words	65.6%	162	Topics	59.6%	216
Persons	59.6%	251	Persons	21.9%	76
Locations	18.9%	50	Locations	19.3%	51
Organizations	2.5%	6	Organizations	9.3%	46
max	1.3%	2	*max*	13.4%	47
average	51.5%	126	*average*	51.8%	114

Table 2: F-measure and # events for the 4-way split vector space and probabilistic models

3.3 Named Entity Recognition

To estimate the influence of the NER, we have manually evaluated performance of NER on a small validation set and found that performance was satisfying: we obtained a precision of 93.37% and a recall of 97.69%. Precision is the percentage of identified person names by the system that corresponds to correct person names, and recall is the percentage of person names in the text that have been correctly identified by the system.

3.4 Results of Event Detection in Wikinews

In this section, we present the performance of our event clustering evaluations. The natural clustering derived from treating a text document as a whole is compared to the one based on splitting the document in different aspects. In the following evaluations, we will list each time the natural clusterings performance for the full text, the different aspects and the different combinations of the aspects.

Topic-entity Our first set of tests evaluates the improvement of event clustering by splitting news stories into two aspects: the topic (words) and the named entities. We test this, both with the vector space model and the probability model as the underlying representations. The results are shown in table 1. In both cases, comparing the full text already give acceptable results (76.8% and 69.5%). The

Vector space	F-measure	# events	*Probabilistic*	F-measure	# events
max	1.3%	2	*max*	13.4%	47
wiki	69.8%	216	*wiki*	41.1%	166
reuters	77.2%	278	*reuters*	47.1%	195
average	51.5%	126	*average*	51.8%	114
wiki	47.4%	109	*wiki*	46.7%	113
reuters	53.7%	130	*reuters*	51.8%	134

Table 3: F-measure and # events when using importance factors for the 4-way split vector space and probabilistic models

separate aspects on their own achieve a lower performance, as we use only part of the content. When combining the aspects' distances using the max-function, we improve on the full text's performance, for both representation models (and especially the vector space model). This shows we can now discriminate stories on a finer level, by comparing more types of information inside the story.

The $average$-function gives an improvement for the probabilistic models, but not for the vector space models. When the original data already gives good results, averaging the aspect distances does not improve document discrimation, as no aspect now has the possibility to discriminate on its own.

Topic-person-location-organization In table 2, we present the results when separating the entities by their semantic classes. When looking at the combination function's results, we see that splitting a document into more aspects vastly decreases performance. Inspection of the different aspects showed this to be caused by sparseness: when a semantic class contains no entities, that aspect's distance to those of other documents is either 0 or 1. In combination with the max-function, this causes the dissimilarity between many documents to be overestimated. The $average$-function suffers less from this problem, as the less sparse aspects can compensate for this behaviour. Its performance however is still too low.

These results show the need for our importance factors: a way to decide when sparse aspects should or should not have an impact in the combination function.

Importance factors In table 3, we compare the results of the max- and $average$-function from table 2 to their weighted versions. The importance factors used to weight the functions have been learned from two separate training corpora (Wikinews and Reuters). As Wikinews is also the test corpus, we essentially bootstrap information that lies in the 2-way split in order to improve the 4-way split. The Reuters corpus is an independent training set, and much larger than the Wikinews corpus.

The results show that the unsupervisedly learned importance factors improve the performance for the max-function significantly. For the $average$-function, they remain comparable. The weights trained from the Reuters-corpus outperform

those trained from Wikinews. The number of training data thus influences the accuracy of the weights.

4 Related research

Assessing similarities and differences between textual documents has a long-standing interest. Established approaches represent documents as term vectors (where terms are possibly weighted by a $tf \times idf$ factor) and compute the cosine of the angle of the term vectors (Salton 1989) (vector space model). These models assume that the vectors that span the geometric space are pairwise orthogonal, an assumption which is violated in real texts (Wang et al. 1992). In order to cope with synonym and related terms, the algebraic vector space model incorporated document representations based on Latent Semantic Indexing (LSI) (Deerwester et al. 1990) and singular value decomposition of the term-by-document matrix of a document collection. Recently LSI models were replaced by probabilistic topic models which deal with polysemy in a more natural way. The main idea is that documents are viewed as a mixture of topics and each topic as a mixture of words. Several latent topic models exist, such as probabilistic Latent Semantic Analysis or pLSA (Hofmann 1999) and Latent Dirichlet Allocation (LDA) (Blei et al. 2003). In both cases the topic and word distributions are learned from a large training corpus, but newer models such as LDA learn additional latent variables that are independent of the training corpus, so that the topic distributions of new, previously unseen documents can be inferred. Variant models have been studied by (Buntine and Jakulin 2006).

Recent work on probabilistic topic models combines metadata content with topic models, as is done by (Mccallum et al. 2005) who steer the discovery of topics according to the relationships between people. These models, although very valuable, add in a limited way some semantics to the words in a document. The document representation however is still a quite rudimentary reflection of its semantics. Structured models that take into account topic correlations have been proposed by (Li and Mccallum 2006). This model did not yet take into account extracted information such as named entities.

In the computational linguistics domain, paraphrasing techniques have been developed in order to detect similar content by considering matching of word co-occurrences, matching noun phrases, verb classes, proper nouns, etc. (Hatzivassiloglou 1998), (Barzilay and Lee 2003), where the matching patterns might be learned in an unsupervised way using sentences that already describe comparable content. As a kind of extension of the paraphrasing models, researchers have attempted to detect contradictions in natural language statements, for instance, by means of handcrafted rules (Mckeown and Radev 1995) or learning contradiction models from annotated sentences (de Marneffe et al. 2008). These techniques are usually confined to finding similarities or differences in a fine grained way, but their use is currently still restricted by a rather low performance, making them less suited for content comparison.

Information extraction technologies that semantically classify certain informa-

tion in the documents (such as named entity recognition) in combination with probabilistic topic models offer many interesting possibilities for representing and comparing texts possibly along different aspects of content. Event detection has received a substantial interest in information retrieval research (often as part of topic detection and tracking (TDT) tasks. Early work on retrospective event detection based on a hierarchical agglomerative clustering (group average clustering) is done by (Yang et al. 1999) (building further on (Cutting et al. 1992)). The events are clustered based on lexical (single words) similarity of the documents and temporal proximity. The temporal proximity parameter avoids clustering documents that are too far apart in time. Many different studies on event detection followed these initial initiatives (see (Allan et al. 2002) for the main approaches). Many of them rely on a vector space representation of the documents, where more recent approaches make a distinction between named entities and non named entity words (e.g., (Kumaran and Allan 2004)). In such a scheme each term type might receive a different weight, possibly learned from a training corpus (Zhang et al. 2007)]. Probabilistic models for representing events in documents are scarce. (Allan et al. 2003) use a simple probabilistic language model as a document representation. (Li et al. 2005) build a probabilistic generative model for retrospective news events detection, where an event generates persons, locations, keywords as named entities apart from a time pointer. Other research on integrating named entities in an event detection task include (Makkonen et al. 2002), (Zhang et al. 2007), where (Zhang et al. 2007) demonstrated correlations between named entity types and news classes.

5 Conclusion

In this paper we presented a comparative study of several unsupervised methods in order to detect similarities and differences between text documents. Our methods were evaluated in the setting of news event clustering. Our main hypothesis was that considering different aspects of documents improves document comparison as a whole. In order to test this hypothesis, we investigated the influence of representation models, both vector space as probabilistic; the influence of the number of aspects to consider; the possible dependencies between aspects; and different methods of combining the aspects information.

The results confirmed our hypothesis. We have shown that regardless the representation models, considering different aspects improves the clustering performance, although there are several restrictions to this claim. Considering too many aspects creates problems of sparseness. Learning dependencies between aspects in an unsupervised way is able to reduce the influence of sparse, irrelevant aspects.

In future work, we like to apply these techniques on other types of texts or media in different comparison or clustering tasks.

References

Allan, James, Courtney Wade, and Alvaro Bolivar (2003), Retrieval and novelty

detection at the sentence level, *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, ACM, New York, NY, USA, pp. 314–321.

Allan, James, Victor Lavrenko, and Russell Swan (2002), *Explorations within Topic Tracking and Detection*, Kluwer Academic Publishers, ir 20, pp. 197–224.

Bagga, Amit and Breck Baldwin (1998), Algorithms for scoring coreference chains, *In The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*, pp. 563–566.

Barzilay, Regina and Lillian Lee (2003), Learning to paraphrase: An unsupervised approach using multiple-sequence alignment, *HLT-NAACL 2003: Main Proceedings*, pp. 16–23.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan (2003), Latent dirichlet allocation, *J. Mach. Learn. Res.* **3**, pp. 993–1022, MIT Press, Cambridge, MA, USA.

Buntine, Wray and Aleks Jakulin (2006), Discrete component analysis, *Subspace, Latent Structure and Feature Selection Techniques*, Springer-Verlag.

Cutting, Douglass R., Jan O. Pedersen, David Karger, and John W. Tukey (1992), Scatter/gather: A cluster-based approach to browsing large document collections, *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 318–329. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.34.6746.

de Marneffe, Marie C., Anna N. Rafferty, and Christopher D. Manning (2008), Finding contradictions in text, *Proceedings of ACL-08: HLT*, Association for Computational Linguistics, Columbus, Ohio, pp. 1039–1047. http://www.aclweb.org/anthology-new/P/P08/P08-1118.bib.

Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman (1990), Indexing by latent semantic analysis, *Journal of the American Society for Information Science* **41**, pp. 391–407.

Hatzivassiloglou, Vasileios (1998), *Automatic acquisition of lexical semantic knowledge from large corpora: the identification of semantically related words, markedness, polarity, and antonymy*, PhD thesis, New York, NY, USA. Adviser-Mckeown,, Kathleen R.

Hofmann, Thomas (1999), Probabilistic latent semantic analysis, *Proceedings of Uncertainty in Artificial Intelligence, UAI*, Stockholm. http://citeseer.csail.mit.edu/hofmann99probabilistic.html.

Kumaran, Giridhar and James Allan (2004), Text classification and named entities for new event detection, *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, pp. 297–304.

Li, Wei and Andrew Mccallum (2006), Pachinko allocation: Dag-structured mixture models of topic correlations, *ICML '06: Proceedings of the 23rd international conference on Machine learning*, ACM, New York, NY, USA, pp. 577–584. http://dx.doi.org/10.1145/1143844.1143917.

Li, Zhiwei, Bin Wang, Mingjing Li, and Wei-Ying Ma (2005), A probabilistic model for retrospective news event detection, *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, pp. 106–113.

Makkonen, Uha, Helena Ahonen-Myka, and Marko (2002), Applying semantic classes in event detection and tracking, *Proc. International Conference on Natural Language Processing (ICON'02)*, pp. 175–183.

Mccallum, Andrew, Andres Corrada-Emmanuel, and Xuerui Wang (2005), Topic and role discovery in social networks, pp. 786–791.

Mckeown, Kathleen and Dragomir R. Radev (1995), Generating summaries of multiple news articles, *In Proceedings, 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 74–82.

Salton, Gerard (1989), *Automatic text processing: the transformation, analysis, and retrieval of information by computer*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Voorhees, Ellen M. (1986), Implementing agglomerative hierarchic clustering algorithms for use in document retrieval, *Technical report*, Ithaca, NY, USA.

Wang, Z. W., S. K. M. Wong, and Y. Y. Yao (1992), An analysis of vector space models based on computational geometry, *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, pp. 152–160.

Yang, Yiming, Jaime G. Carbonell, Ralf D. Brown, Thomas Pierce, Brian T. Archibald, and Xin Liu (1999), Learning approaches for detecting and tracking news events, *IEEE Intelligent Systems* **14** (4), pp. 32–43, IEEE Computer Society, Los Alamitos, CA, USA.

Zhang, Kuo, Juan Zi, and Li Gang Wu (2007), New event detection based on indexing-tree and named entity, *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, USA, pp. 215–222.