Department of Creative Informatics
Graduate School of Information Science and Technology
THE UNIVERSITY OF TOKYO

Master Thesis

Clustering multilingual documents by estimating text-to-text semantic relatedness

文書間の意味的関係性の推定に基づく多言語文書クラスタリング

Dani Yogatama
ダニ・ヨガタマ

Supervisor: Associate Professor Kumiko Tanaka-Ishii

August 2010

For S.W.P.

Abstract

This thesis is about multilingual document clustering through estimating semantic relatedness between multilingual texts. Specifically we focus on the task of clustering multilingual documents with very limited or no supervisory information. We present two approaches to address the problem : a comparable-corpora based approach and a web-searches based approach. Our first approach derives pairwise constraints from comparable corpora and cluster multilingual documents in a semi-supervised manner. The method models document collections as weighted graph, and supervisory information is given as sets of must-link constraints for documents in different languages. Recursive k-nearest neighbor similarity propagation is used to exploit the prior knowledge and estimate semantic relatedness of multilingual documents. Spectral method is then applied to find the best cuts of the graph. Our second approach to multilingual document clustering uses web-searches to estimate semantic relatedness between multilingual words. In this approach, we extract informative terms from each language involved and query a search engine using each pair of extracted terms as keywords to construct a web-count based word similarity matrix. A variant of hierarchical agglomerative clustering algorithm is then applied to discover multilingual word clusters, and the resulting word clusters are utilized as features to perform document clustering. Evaluation of experimental results using various evaluation measures suggests that the proposed algorithms achieve satisfactory clustering result and outperforms existing methods which utilize similar supervisory information. Furthermore, since we do not use any language dependent information in the clustering process, our algorithm can be applied to documents which are written in different writing systems, such as Japanese and English texts.

概要

本論文は文書間の意味的関係性の推定に基づく多言語文書クラスタリング、特に事前情報の与えられていない教師なしの多言語文書クラスタリングについて述べる。我々は対訳コーパスに基づく手法と情報検索に基づく手法の、二つの手法を提案する。一つ目は対訳コーパスから制約を作成し、多言語文書の半教師ありクラスタリングを行う。提案手法は文書集合を重み付きグラフとしてモデル化する。異なる言語の文書を扱うために、must-link による制約として対応する文書の情報を与える。反復的な k-近傍 similarity propagation を用いて、多言語を含む言語空間を作成する。スペクトラルな手法を用いて最適なグラフカットを求める。二つ目は情報検索システムを利用し、多言語の単語間における意味的関係性を推定する手法である。本手法は各言語において検索の際に重要な単語を抽出し、全ての単語ペアをクエリとした検索を行うことによって、単語間の類似度行列を作成する。Hierarchical Agglomerative Clustering アルゴリズムを用いて多言語を含んだ単語クラスタを作成する。この単語クラスタを文書クラスタリングに素性として利用するのが特徴である。評価実験において二つの提案手法は適切な文書クラスタを作成し、既存手法に比べて優位な結果が得られた。各言語に依存する情報を用いないことから、本手法は日本語や英語といった様々な言語に対して応用可能である。

Acknowledgements

First, I would like to thank Professor Kumiko Tanaka-Ishii, for her guidance, support, and encouragement during the course of my masters study. Thank you for introducing me to academic life and teaching me invaluable life lessons. I am truly honored to have had you as an advisor.

This thesis would have not been possible without the generous grants of Ministry of Education, Culture, Sports, Science and Technology, Government of Japan. I was supported by the Japanese Goverment Monbukagakusho Scholarship for my entire study at the University of Tokyo from 2008 to 2010.

I am also grateful for my colleagues in Tanaka laboratory, who have made my time in Japan a wonderful experience. Special thanks to : Satoshi Tezuka, Kotaro Kitagawa, Takahiro Ando, and my tutor Tei Tou.

Finally, I thank my family, my parents and my brothers, for giving me unconditional love and supporting me in every aspect of my life. I am blessed to have you in my life.

Contents

Chapter 1

Introduction

1.1 Multilingual document clustering

Document clustering is unsupervised classification of text collections into distinct groups of similar documents, where similarity is defined as some function on documents. Figure 1.1 shows an example of logical grouping of documents by a clustering algorithm. Generally, a document clustering algorithm partitions documents based on their topic similarities. This means that we expect to have documents which discuss the same topic assigned to a single cluster.

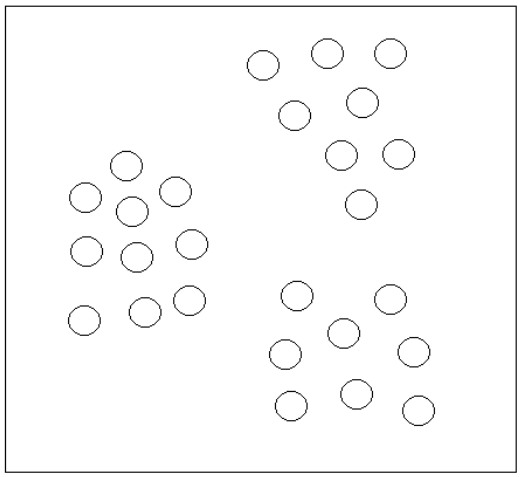
Document clustering should not be confused with text classification. Similar to document clustering, text classification also involves partitioning documents into cohesive groups. However, the key difference between document clustering and text classification is that clustering is an unsupervised learning, while classification is a supervised learning. Therefore, in text classification, we have a set of training documents manually assigned to their respective classes by human expert, and the algorithm tries to replicate this categorization by learning from the training data. On the other hand, in document clustering, the categorical distinction is not known beforehand so the clustering process involves discovering these categories.

Application of document clustering goes beyond data organization, i.e., organizing collection of documents into knowledge maps [27, 20]. It has been used in many information retrieval tasks such as cluster-based retrieval [26] and cluster-based browsing (scatter-gather) [7, 30]. Furthermore, it has also been applied as a preprocessing step in natural language processing tasks including language modeling [17] and improving the performance of a text categorization system [1].

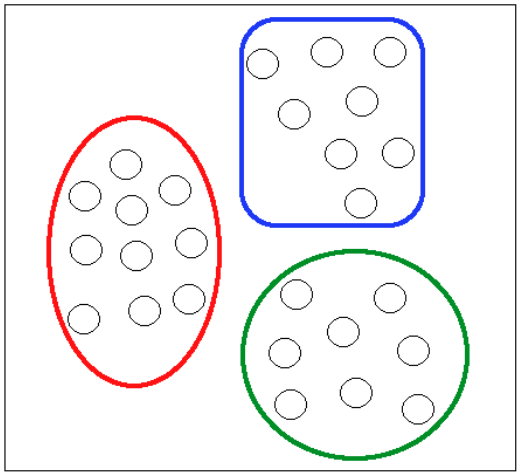
In order to discover high-quality clusters, it is important to define a good similarity function on which the clustering algorithm is built on. Cosine similarity measure based on vector space model document representation is the most commonly used similarity function. In this measure, each document is represented as a feature vector whose dimension is equal to the number of document attributes in the collection. The attributes can be chosen as words, subset of words, phrases, etc.; and the weight can be binary, a function of the frequency of occurrence in the document (tf), a function of the frequency of occurrence in the entire collection (idf), or the combinations of them.

Various document clustering techniques have been proposed, but most of them deal with monolingual documents. Advances in internet technology have resulted in an insurmountable growth of number of multilingual documents available on the Web. These documents are written in various languages on diverse topics, and the task of organizing them becomes a critical problem of knowledge management practice. There exist thousands of languages, not to mention countless minor ones. Creating document clustering model for each language is simply unfeasible. The demand for a robust multilingual document clustering algorithm is rapidly increasing, since we need methods to deal with text collections in diverse languages simultaneously.

Multilingual document clustering involves partitioning documents, written in more than one language, into sets of clusters. In multilingual document clustering, similar documents, even if



(a) Document collections



(b) Document clusters

Fig. 1.1. Example of discovering partition of document collections. It is visually clear that there are three distinct clusters of documents. A document clustering algorithm is expected to discover these clusters in an unsupervised manner.

they are written in different languages, should be grouped together into one cluster. It is straightforward to understand that a monolingual document clustering algorithm cannot be used since the similarity of multilingual documents cannot be computed using a standard model, because documents in different languages are mapped into different spaces. Therefore, the major challenge of multilingual document clustering is computing the similarity between multilingual documents.

Multilingual document clustering is related to multilingual text categorization and cross-language information retrieval. These three problems requires the notion of similarity between multilingual texts. In cross-language information retrieval, we have to compute the similarity between user query and documents in the collection, and return documents which are similar to the query but are written in different language. Multilingual text classification is even more related

to multilingual document clustering, since we usually can adjust an algorithm for multilingual text classifiation to handle clustering problem, and vice versa. Our main focus in this thesis is the estimation of semantic relatedness for comparing documents in different languages with application to document clustering. Though the method can be adapted to handle multilingual text classification or even cross-language information retrieval, we shall proceed with document clustering as the main application in mind.

1.2 Semantic relatedness

Semantic relatedness defines how related two entities are based on their meaning (semantic). The term semantic relatedness extends beyond synonymy. It includes antonymy, meronymy, hyponymy, etc. For example, the word `finger` is highly related to the word `hand`, even though they do not explain the same object. `Finger` and `hand` is an example of meronymy, since a finger is part of a hand. Other examples of semantically related words include `war` and `peace` (antonym), `pigeon` and `bird` (hyponym), as well as `notebook` and `laptop` (synonym). Measuring semantic relatedness between words is particularly useful for disambiguating word sense, automatically building thesaurus, and extracting synonyms from a collection of texts.

There are many automatic measures to estimate semantic relatedness between words, Latent Semantic Analysis[*1] (LSA) being one of the most prominent of them. It computes the similarity between concepts based on their co-occurrences. Other methods use search engine to compute web-count based similarity score. These methods usually combine web-count results with text snippets returned by a Web search engine to construct feature vectors for computing the similarity score [4, 5, 6]. Information about text snippets is usually incorporated in the feature vector to capture context information of a particular pair of words. Lexical ontologies such as WordNet has also been used for estimation of semantic relatedness [2, 3]. In fact, the WordNet itself groups semantically related words (i.e. synsets) to a single concept.

In the context of document clustering, and throughout this thesis, we consider two entities as semantically related if they belong to the same topic. Since this makes semantic relatedness almost similar to similarity, we will use the terms interchangeably throughout this thesis. While it is logical to assume that an entity is a word, it can also be a set of words or even a document. For example, in our first approach the entities are documents, and we use a coarse approximation method when measuring semantic relatedness between them by proposing a technique called similarity propagation. We can consider a document about research in biology as semantically related to a physics article, since they both belong to the same topic (i.e. science), even if they are written in different languages[*2]. A good indication of relatedness between monolingual documents is the number of common words in those documents. However, since multilingual words are written differently, in this approach we bypass word-level representation and operate directly in document-level representation when estimating semantic relatedness between multilingual texts.

Our second approach is different since it uses a two-level approximation when estimating semantic relatedness between documents. First, the algorithm measures semantic relatedness between multilingual words. For example, the word `missile` is related to the word **戦争**(war) since they are indicative of the topic *war*. Next, the algorithm clusters the words and uses these results to estimate semantic relatedness between multilingual texts.

We shall go back to this and describe how each approach estimates semantic relatedness between texts in details in Chapter 2 and Chapter 3.

[*1] Latent Semantic Analysis is sometimes referred to as Latent Semantic Indexing (LSI).

[*2] We assume that the algorithm is not required to differentiate documents in topic science into subtopics.

1.3 Previous work on multilingual document clustering

Early works on multilingual document clustering use dictionary, statistical machine translation, or multilingual ontology to achieve cross-lingual semantic interoperability. Chen et al. [6] proposed a two-level approach to cluster multilingual documents using translation technology by relying on cross-lingual dictionary. In their method, monolingual documents are classified according to a predefined topic set in their respective language first, before alignments between multilingual clusters are mined to match the topics using a dictionary. Verbs, named entities, and nouns are used to measure the similarity of two multilingual clusters when matching them. If there are more than one translation for a word in dictionary, the most frequent translation is chosen as the "correct" translation. Since a monolingual cluster can have no matching cluster in other language, some of the resulting clusters might be purely monolingual.

Evans and Klavans [11] presented a method which uses dictionary lookup to translate all documents in Japanese and Russian to English. Similar to [6], no word sense disambiguation is performed. A machine translation system is used to translate documents in languages other than Russian and Japanese to English. The reason behind the usage of dictionaries for Japanese and Russian and machine translation for others is that they have created a fast technique for dictionary lookup in Japanese and Russian, but not for other languages. After translation is performed, the method proceeds by considering all documents as written in uniform language and applies a general monolingual algorithm to discover document clusters.

Pouliquen et al. [25] map multilingual documents to a multilingual thesaurus of European languages called Eurovoc[*3] to calculate cross-lingual document similarity. Eurovoc is a multilingual ontology covering various domains such as politics, economics, science, industry, etc. It exists in 23 European languages, making it suitable to estimate similarity of documents in various European languages. They proposed a method to produce an automatically generated overview of news collections in English, German, French, Spanish, and Italian, by clustering multilingual news which belong to the same topic. To carry out the task, they first extract features from document collections as well as identify place name, before using Eurovoc to map documents to a multilingual classification scheme. The mapping can be described as a category ranking classification task, which produced long ranked lists of relevant classes for each document [25]. This new language independent representation is then used for calculating similarity between multilingual documents.

More recently, parallel texts and comparable corpora have been used to build a multilingual clustering space. Comparable corpora are collections of texts in different languages regarding similar topics which are produced at the same period. The key difference between comparable corpora and parallel texts is that documents in comparable corpora are not necessarily translations of each other. However, terms in comparable corpora, taken from documents in similar topic, often describe the same concept in different languages. While comparable corpora provide weaker supervisions than parallel texts, they are considerably easier to be acquired and prepared.

Wei et al. [29] use LSA-based approach to utilize parallel texts in multilingual document clustering. First, they extract informative terms (particularly nouns and noun phrases) from each parallel documents in the corpus. Next, they create a term-by document matrix using the extracted terms and collapse columns whose documents are translations of each other. Subsequently, LSA is applied to the matrix to construct a multilingual semantic space and reduce the dimensions of document vectors. To cluster a new set of documents, it is projected to the multilingual semantic space, and a hierarchical clustering algorithm is then applied to discover clusters of these multilingual documents in the language-independent space.

[*3] http://europa.eu/eurovoc/

Gliozzo and Strapparava [13] also applies LSA to perform cross-language text categorization by building multilingual domain models from comparable corpora. They rely on the presence of common words and proper nouns across various languages to build multilingual domain models, which are used to define a generalized similarity function between documents in different languages. A domain model consists of soft clusters of terms, and each cluster can be considered as a semantic domain. The terms in a cluster belong to the same semantic field and thus are highly related. LSA is used to automatically create this multilingual domain models from comparable corpora. Specifically, they consider the resulting LSA dimensions as multilingual clusters of terms and documents, and use them to estimate similarity among texts in different languages. Once similarity between multilingual documents is computed, they train a multilingual domain kernel and use Support Vector Machines to perform cross language text categorization. Zhang and Mao [31] used a related technique called Modularity Eigenmap to extract community structure features from the document network to solve hypertext classification problem.

In cross-language information retrieval, a closely related field to multilingual document clustering, Dumais et al. [10] presents a method to retrieve collection of documents in language different than that of the query by using LSA.

Table 1.1 provides summary of previous work on multilingual document clustering.

Supervisory information	**Method**
Dictionary	Align multilingual clusters [6]
Dictionary	Translate documents into single language [11]
Multilingual thesaurus	Map documents to language independent representation using thesaurus [25]
Parallel texts	Merge parallel documents and perform Latent Semantic Analysis [29]
Comparable corpora	Latent Semantic Analysis to create Multilingual Domain Models [13]

Table. 1.1. Summary of previous work on multilingual document clustering.

1.4 Challenges of multilingual document clustering

While clustering in its purest form is a completely unsupervised task, existing multilingual document clustering techniques require the presence of supervisory information (i.e., dictionary, multilingual thesaurus, parallel texts, or comparable corpus) to achieve cross-lingual semantic interoperability, as described in the previous section. In this section, we shall see limitations of each supevisoy information which hinder the progress of multilingual document clustering.

Dictionaries, particularly those between major languages such as English - Japanese or French - Japanese, are prevalent, while multilingual thesauruses are significantly harder to be acquired. However, both suffer the same problems when they are used as supervisory information for multilingual document clustering. First, even a very large dictionary does not cover entire vocabulary of that particular language, since new words such as proper names are created almost on a daily basis. Untranslated words might bias a document clustering algorithm to prefer documents in the same language since they add to the similarity measure. Word ambiguity also carries the potential to disrupt a clustering algorithm since usually the algorithm only picks the most frequent sense and ignores the others. A multilingual thesaurus, besides having the same drawbacks as dictionary, is expensive to be built. Existing multilingual thesaurus which is regularly used to for multilingual document clustering is Eurovoc. It is only available in 22 official languages of the European Union (Bulgarian, Spanish, Czech, Danish, German, Estonian, Greek, English, French, Italian, Latvian, Lithuanian, Maltese, Hungarian, Dutch, Polish, Portuguese, Romanian, Slovak, Slovene, Finnish and Swedish) and Croatian. While such resource is highly valuable, a multilingual document clustering algorithm based on it cannot be generalized to other languages without

adding the languages to Eurovoc or creating an entirely new multilingual thesaurus.

Recent success in statistical machine translation in using parallel texts as supervisory information makes them highly attractive for multilingual document clustering. It seems logical to collapse documents that are translations of each other in a term-by-document matrix to create a multilingual document space. However, similar to the problem of the utilization of parallel texts in statistical machine translation, they only exist for a few major languages. The cost of creating ones is high, and currently only few websites maintain parallel documents in several languages. Moreover, the texts only cover limited domains, and several efforts to use them as supervisory information for documents in different domains still fail to produce satisfactory results.

Unlike parallel texts, comparable corpora are easier to be acquired. News agencies often give information in many different languages and can be good sources for comparable corpora. Comparable corpora-based multilingual document clustering generally uses spectral method such as LSA to collect information about the correlation of multilingual words. Current techniques strongly rely on the presence of common words across different languages. They exploit proper nouns representing unique entities such as place name, person name, organization name, etc., which are written uniformly across languages. However, this method would only work if the languages are highly related, i.e., languages in the same family. In languages from different families, place names or person names are often transliterated differently. For example, the country `Japan` is written `Jepang` in Indonesian and 日本 in Japanese. Existing comparable-corpora based document clustering algorithms fail to discover good multilingual clusters for documents in unrelated languages due to this problem, making it difficult to generalize their usages to heterogeneous languages.

Table 1.2 provides summary of limitations of each supervisory information in previous work on multilingual document clustering.

Supervisory information	**Challenge**
Dictionary	Word ambiguity, dictionary limitation
Multilingual thesaurus	Word ambiguity, thesaurus limitation, thesaurus availability
Parallel texts	Parallel texts availability
Comparable corpora	Reliance on common words across languages

Table. 1.2. Limitations of each supervisory information in previous work on multilingual document clustering.

1.5 Overview of this thesis

The center point of this thesis is the problem of clustering multilingual documents into distinct sets of groups based on their topic similarities. We have shown several limitations faced by a multilingual document clustering algorithm in the previous section. The goal of this thesis is to introduce robust clustering methods which can be applied to documents in various languages. We attack the problem of multilingual document clustering by trying to estimate semantic relatedness between multilingual texts so that we can compute similarity measure between those documents. We propose two methods in this thesis, both are language and domain independent, as well as able to handle documents in different writing systems such as English and Japanese.

In chapter 2, we present a graph-based approach to multilingual document clustering using comparable corpora. In our first method, we try to utilize prior knowledge in the form of must-link constraints, gathered from comparable corpora, to estimate text-to-text semantic relatedness. Unlike existing comparable corpora based algorithms, our method does not use LSA. It considers documents in the same topic as pairwise constraints and performs similarity propagation to other

documents while constructing a graph of multilingual documents. The multilingual clusters are then discovered by finding the best partition of the graph. Semantic relatedness estimation method called similarity propagation is used to guide the language-space merging process.

In chapter 3, we show a method to cluster multilingual documents without using traditional supervisory information (dictionary, parallel texts, comparable corpora, etc.), but by performing web-searches to estimate similarity between multilingual documents using semantically related word clusters. Our major contribution here is twofold. We present a multilingual word clustering algorithm, and further use it as an intermediate step in the proposed document clustering method. Moreover, the word clustering algorithm also has the potential to be used for mining alignment from comparable corpora, as shown by our experimental results.

The last chapter concludes this thesis with a summary and direction for future works.

Chapter 2

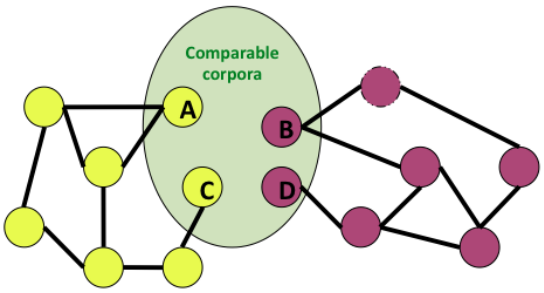
Multilingual Document Clustering Using Comparable Corpora

In this chapter, we present our first clustering algorithm which uses comparable corpora as supervisory information to estimate semantic relatedness between multilingual documents. As mentioned in the previous chapter, our clustering model can be applied to any multilingual text collection regardless of the languages, providing that comparable corpora in those languages exist. The method models document collections as a weighted graph and takes supervisory information in the form of must-link constraints for documents in different languages. Recursive k-nearest neighbor similarity propagation is used to utilize the prior knowledge and merge multilingual spaces. The algorithm can be considered as a spectral clustering method since it uses the information contained in eigenvectors of a normalized document similarity matrix derived from the data to find clusters. Spectral clustering algorithm that works in monolingual context has been proposed in [24]. The paper also contains an in-depth analysis of spectral algorithm for clustering problems. Spectral clustering has also been applied to other applications such as information retrieval [8] and computer vision [21]. Our major contribution here is the propagation method to make spectral clustering algorithm works for multilingual problems.

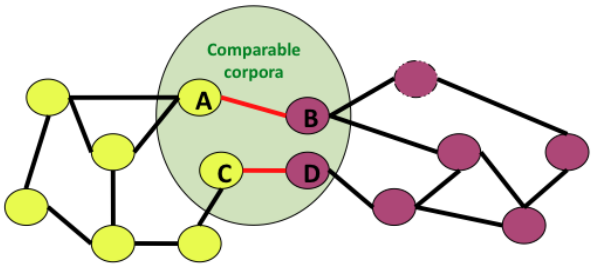
It is worth mentioning that our method is not the first spectral method for multilingual document clustering, since LSA is also a spectral method. However, in previous works, it is mainly used to exploit the presence of common words and proper nouns across languages in comparable corpora, whereas our algorithm uses the spectral method to find the best partition of document collection graph. Therefore, our method does not suffer the same problem as existing approaches using comparable corpora, i.e. the performances of those methods highly depend on the number of common words in languages involved.

Since we use pairwise constraints from comparable corpora to estimate similarity between multilingual documents, the method can also be considered as a semi-supervised clustering algorithm. In general, semi-supervised clustering enhances clustering task by incorporating prior knowledge to aid clustering process. It allows user to guide the clustering process by giving some feedbacks to the model. In unsupervised clustering algorithm, only unlabeled data is used to find assignments of data points to clusters. In semi-supervised clustering, prior knowledge is given to improve performance of the system. The supervision can be given as must-link constraints and cannot-link constraints, first introduced in [28]. Note that we only utilize must-link constraints in our algorithm. Kamvar et al. [14] proposed spectral learning algorithm that can take supervisory information in the form of pairwise constraints or labeled data. Their algorithm is intended to be used in monolingual context, while our algorithm is designed to work in multilingual context.

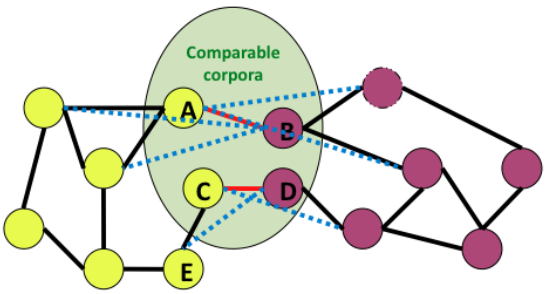
We formulate the problem of multilingual document clustering as a spectral clustering task in the next section, and present our algorithm in the subsequent section. We conclude the chapter with experimental results and discussions.



(a) Document collection is combined with comparable corpora and modeled as undirected graph. Each node represents a document and the weight of each edge is defined as the cosine similarity between two adjacent nodes (documents).

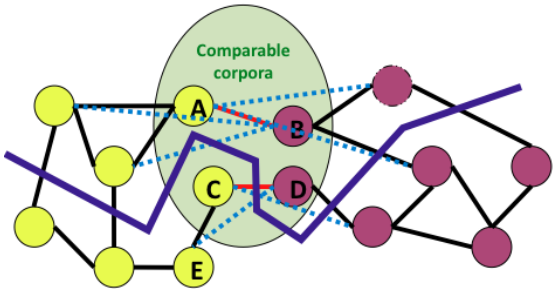


(b) Suppose that document A - B and C - D belong to the same topic in our comparable corpora, we create must-link constraints (red lines) between these documents and assign the weight of these edges to one.



.

(c) Propagate similarity to neighboring nodes to create more edges (blue lines) as described in Section 2.3 (estimation of text-to-text semantic relatedness)



(d) Use spectral clustering algorithm to find the best partition of the graph.

Fig. 2.1. Illustration of our multilingual spectral clustering algorithm.

2.1 Problem definition

We combine documents to be clustered and comparable corpora to get the document collection of our algorithm. The document collection is being modeled as undirected graph $G(V, E, W)$, where V, E, and W denote the graph vertex set, edge set, and transition probability matrix, respectively. In graph G, $v \in V$ represents a document, and weight $w_{ij} \in W$ represents transition probability between document v_i to v_j. The transition probabilities can be interpreted as edge flows in Markov random walk over graph vertices (documents in collection), and are computed using cosine similarity measure. Overall algorithm is given in Section 1 and the method to estimate semantic relatedness of multilingual texts by similarity propagation is given in Section 2. Figure 2.1 shows an illustration of our document clustering algorithm.

Algorithm 1 Multilingual Spectral Clustering

Input: Term by document matrix M, pairwise constraints
Output: Document clusters

1: Create graph affinity matrix $A \in \mathbb{R}^{n \times n}$ where each element A_{ij} represents the similarity between document v_i and v_j.
2: **for all** pairwise constraints in comparable corpora **do**
3: $A_{ij} \leftarrow 1, A_{ji} \leftarrow 1$.
4: Recursive Propagation $(A, S, \beta, k, v_i, v_j)$.
5: **end for**
6: Post-process matrix A so that every value in A is greater than δ and less than 1.
7: Form a diagonal matrix D, where $D_{ii} = \sum_j A_{ij}$. Normalize $N = D^{-1}A$.
8: Find $x_1, x_2 \cdots, x_t$, the t largest eigenvectors of N.
9: Form matrix $X = [x_1, x_2, \cdots, x_t] \in \mathbb{R}^{n \times t}$.
10: Normalize row X to be unit length.
11: Project each document into eigen-space spanned by the above t eigenvectors (by treating each row of X as a point in $\mathbb{R}^t$, row i represents document v_i).
12: Apply K-means algorithm in this space to find document clusters.

2.2 Spectral Clustering Algorithm

Algorithm to perform mutlingual spectral clustering is given in Algorithm 1. Figure 2.1 illustrates the clustering process of our method. Let A be affinity matrix where element A_{ij} is cosine similarity between document v_i and v_j (Algorithm 1, line 1). It is straightforward that documents belonging to different languages will have similarity zero. Rare exception occurs when they have common words because the languages are related one another. As a consequence, the similarity matrix will have many zeros. Our model amplifies prior knowledge in the form of comparable corpora by performing document similarity propagation, presented in Section 2.3 (Algorithm 1, line 4; Algorithm 2, explained in Section 2.3). After propagation, the affinity matrix is post-processed (Algorithm 1, line 6, explained in Section 2.3) before being transformed into transition probability matrix.

The transformation can be done using any normalization for spectral methods. Define $N = D^{-1}A$, as in [22], where D is the diagonal matrix whose elements $D_{ij} = \sum_j A_{ij}$ (Algorithm 1, line 7). Alternatively, we can define $N = D^{-1/2}AD^{-1/2}$ [24], or $N = (A + d_{max}I - D)/d_{max}$ [12], where d_{max} is the maximum rowsum of A. For our experiment, we use the first normalization method, though other methods can be applied as well.

[22] show that probability transition matrix N with t strong clusters will have t piecewise constant eigenvectors. They also suggest using these t eigenvectors in clustering process. We use the information contains in t largest eigenvectors of N (Algorithm 1, line 8-11) and perform K-means clustering algorithm to find the data clusters (Algorithm 1, line 12).

2.3 Similarity Propagation

We use information obtained from comparable corpora to estimate semantic relatedness and define similarity of multilingual texts. Suppose we have text collections in L different languages. We combine this collections with comparable corpora, also in L languages, that act as our supervisory information. Comparable corpora are used to gather prior knowledge by making must-link constraints for documents in different languages that belong to the same topic in the corpora. Our clustering model exploits the supervisory information by detecting k nearest neighbors of the newly-linked documents, and propagates document similarity to these neighbors.

Initially, our affinity matrix A represents cosine similarity between all pairs of documents. A_{ij} is set to zero if j is not the top k nearest neighbors of i and likewise. Next, set A_{ij} and A_{ji} to 1 if document i and document j are different in language and belong to the same topic in our comparable corpora. This will incorporate the must-link constraint to our model. We can also give supervisory information for pairs of document in the same language, but this is optional. We also do not use cannot-link constraints since the main goal is to merge multilingual spaces. In our experiment we show that using only must-link constraints with propagation is enough to achieve encouraging clustering results.

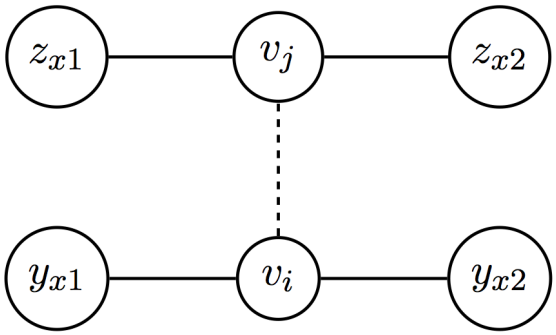
The supervisory information acquired from comparable corpora only connects two nodes in our graph. Therefore, the number of edges between documents in different languages is about as many as the number of must-link constraints given. We argue that we need more edges between pairs of documents in different languages to get better results.

We try to build more edges by propagating similarity to other documents that are most similar to the newly-linked documents. Figure 2.2 gives an illustration of edge-creation process when two multilingual documents (nodes) are connected. Suppose that we have six documents in two different languages. Initially, documents are only connected with other documents that belong to the same language. The supervisory information tells us that two multilingual documents v_i and v_j should be connected (Figure 2.2(a)). We then build an edge between these two documents. Furthermore, we also use this information to build edges between v_i and neighbors of v_j and likewise (Figure 2.2(b)).

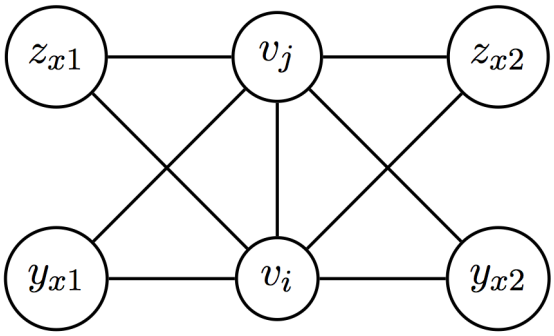
This follows from the hypothesis that bringing together two documents should also bring other documents that are similar to those two closer in our clustering space. [15] stated that a good clustering algorithm, besides satisfying known constraints, should also be able to satisfy the implications of those constraints. Here, we allow not only instance-level inductive implications, but utilize it to get higher-level inductive implications. In other words, we alter similarity space so that it can detect other clusters by changing the topology of the original space.

The process is analogous to shortening the distance between sets of documents in Euclidean space. In vector space model, two documents that are close to each other have high similarity, and thus will belong to the same cluster. Pairing two documents can be seen as setting the distance in this space to 0, thus raising their similarity to 1. While doing so, each document would also draw sets of documents connected to it closer to the centre of the merge, which is equivalent to increasing their similarities.

Suppose we have document v_i and v_j, and y and z are sets of their respective k nearest neighbors, where $|y| = |z| = k$. The propagation method is a recursive algorithm with base S, the number of desired level of propagation. Recursive k-nearest neighbor makes decision to give high similarity between multilingual documents not only determined by their similarity to the newly-



(a) Connect two nodes



(b) Effect on neighbor nodes

Fig. 2.2. Pairing two multilingual documents affect their neighbors. v_i and v_j are documents in two different languages. y_x and z_x are neighbors of v_i and v_j respectively.

linked documents, but also their similarity to the k nearest neighbors of the respective document. Several documents are affected by a single supervisory information. This will prove useful when only limited amount of supervisory information given. It uses document similarity matrix A, as defined in the previous section.

1. For $y_x \in y$ we propagate $\beta A_{v_i y_x}$ to $A_{v_j y_x}$. Set $A_{y_x v_j} = A_{v_j y_x}$ (Algorithm 2, line 5-6). In other words, we propagate the similarity between document v_i and y nearest neighbors of v_i to document v_j.
2. Similarly, for $z_x \in z$ we propagate $\beta A_{v_j z_x}$ to $A_{v_i z_x}$. Set $A_{z_x v_i} = A_{v_i z_x}$ (Algorithm 2, line 10-11). In other words, we propagate the similarity between document v_j and z nearest neighbors of v_j to document v_i.
3. Propagate higher order similarity to k nearest neighbors of y and z, discounting the similarity quadratically, until required level of propagation S is reached (Algorithm 2, line 7 and 12).

The coefficient β represents the degree of enforcement that the documents similar to a document in one language, will also have high similarity with other document in other language that is paired up with its ancestor. On the other hand, k represents the number of documents that are affected by pairing up two multilingual documents. After propagation, similarity of documents that falls below some threshold δ is set to zero (Algorithm 1, line 6). This post-processing step is performed to nullify insignificant similarity values propagated to a document. Additionally, if there exists similarity of documents that is higher than one, it is set to one.

Algorithm 2 Recursive Propagation

Input: Affinity matrix A, level of propagation S, β, number of nearest neighbors k, document v_i and v_j

Output: Propagated affinity matrix

if $S = 0$ **then**
  return
else
  for all $y_x \in k$-NN document v_i **do**
    $A_{v_j y_x} \leftarrow A_{v_j y_x} + \beta A_{v_i y_x}$
    $A_{y_x v_j} \leftarrow A_{v_j y_x}$
    Recursive Propagation $(A, S-1, \beta^2, k, y_x, v_j)$
  end for
  for all $z_x \in k$-NN document v_j **do**
    Set $A_{v_i z_x} \leftarrow A_{v_i z_x} + \beta A_{v_j z_x}$
    Set $A_{z_x v_i} \leftarrow A_{v_i z_x}$
    Recursive Propagation $(A, S-1, \beta^2, k, v_i, z_x)$
  end for
end if

2.4 Experiments

The goals of empirical evaluation include (1) testing whether the propagation method can merge multilingual space and produce acceptable clustering results; (2) comparing the performance to spectral clustering method without propagation.

2.4.1 Data Description

We tested our model using Reuters Corpus Volume 2 (RCV2), a multilingual corpus containing news in thirteen different languages. For our experiment, three different languages: English, French, and Spanish; in six different topics: science, sports, disasters accidents, religion, health, and economy are used. We discarded documents with multiple category labels. We also tested the method on a second dataset consisting of English and Japanese documents in three topics : war and violence, weather, and economy. Since Japanese text does not have inter-word markers, we first parsed the original document using MeCab [16] to get its word-level representation.

We do not apply any language specific pre-processing method to the raw text data. Monolingual TFIDF is used for feature weighting. All document vectors are then converted into unit vector by dividing by its length. Table 2.2 and 2.4 shows the average length of documents in our corpus.

2.4.2 Evaluation Metric

For our experiment, we used Rand Index (RI) which is a common evaluation technique for clustering task where the true class of unlabeled data is known. Rand Index measures the percentage of decisions that are correct, or simply the accuracy of the model. Rand Index is defined as:

$$RI = \frac{TP + TN}{TP + FP + TN + FN}$$

	English	French	Spanish	Total
Science	100	100	100	300
Sports	100	100	100	300
Disasters	100	100	100	300
Religion	100	100	100	300
Health	100	100	100	300
Economy	100	100	100	300
Total	600	600	600	1800

Table. 2.1. Number of documents in the first dataset.

	English	French	Spanish	Total
Science	290.10	165.10	213.45	222.88
Sports	182.55	156.83	189.75	176.37
Disasters	154.29	175.89	165.31	165.16
Religion	317.77	177.91	242.67	246.11
Health	251.19	233.70	227.25	237.38
Economy	266.89	192.55	306.11	255.08
Total	243.79	183.61	224.09	217.16

Table. 2.2. Average number of words of documents in the first dataset.

	English	Japanese	Total
War	392	235	627
Weather	468	468	936
Economy	326	390	716
Total	1186	1093	2279

Table. 2.3. Number of documents in the second dataset.

	English	Japanese	Total
War	300.45	128.29	235.92
Weather	205.63	125.35	165.49
Economy	210.12	143.95	174.08
Total	295.95	132.62	217.62

Table. 2.4. Average number of words of documents in the second dataset.

Rand Index penalizes false positive and false negative decisions during clustering. It takes into account decision that assign two similar documents to one cluster (TP), two dissimilar documents to different clusters (TN), two similar documents to different clusters (FN), and two dissimilar documents to one cluster (FP). We do not include links created by supervisory information when calculating true positive decisions and only consider the number of free decisions made.

We also used F_α-measure, the weighted harmonic mean of precision (P) and recall (R). F_α-

measure is defined as:

$$F_\alpha = \frac{(\alpha^2+1)PR}{\alpha^2 P + R}$$
$$P = \frac{TP}{TP+FP}$$
$$R = \frac{TP}{TP+FN}$$

Last, we used purity to evaluate the accuracy of assignments. Purity is defined as:

$$Purity = \frac{1}{N}\sum_t \max_j |\omega_t \cap c_j|$$

where N is the number of documents, t is the number of clusters, j is the number of classes, ω_t and c_j are sets of documents in cluster t and class j respectively.

2.5 Results and Discussions

We first tested our algorithm on four topics, science, sports, religion, and economy. We then tested our algorithm using all six topics to get an understanding of the performance of our model in larger collections with more topics. We used subset of our data as supervisory information and built must-link constraints from it. The number of must-link constraints provided to the system is given in x-axis (Figure 2.3 - Figure 2.5). Since the number of documents in each language for our experiment is the same, we have the same numbers of documents in subset of English collection, subset of French collection, and subset of Spanish collection. We also ensure there are same numbers of documents for a particular topic in all three languages. We can build must-link constraints as follows. For each document in the subset of English collection, we create must-link constraints with one randomly selected document from the subset of French collection and one randomly selected document from the subset of Spanish collection that belong to the same topic with it. We then create must-link constraint between the respective French and Spanish documents. The constraints given to the algorithm are chosen so that there are several links that connect every topic in every language. Note that the class label information is only used to build must-link constraints between documents, and we do not assign the documents to a particular cluster.

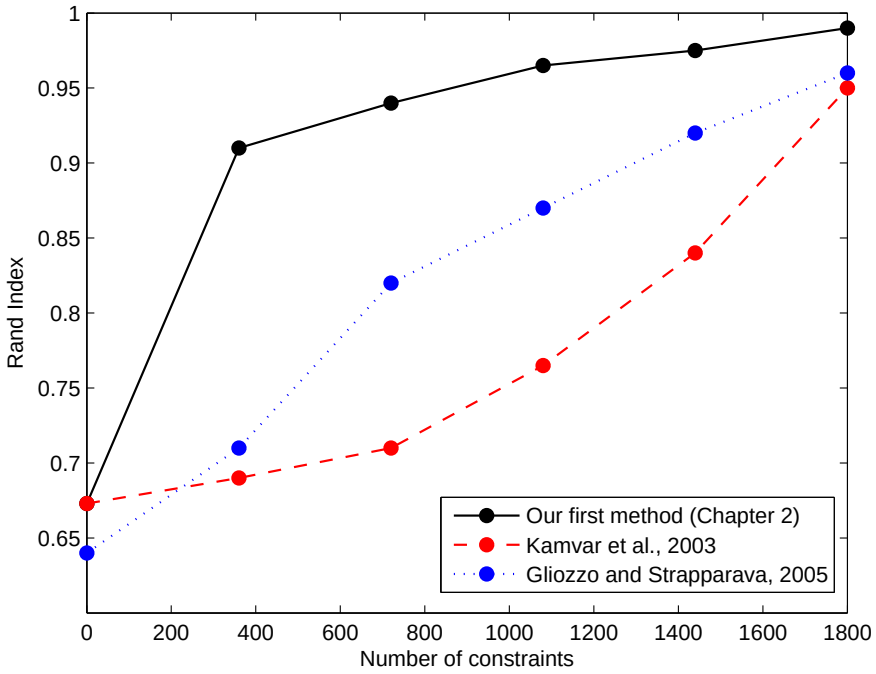
Figure 2.3 shows the Rand Index as the number of constraints increases. Figure 2.4 and Figure 2.5 give purity and F_2-measure for the algorithm respectively. To show the importance of the propagation in multilingual space, we give comparison with spectral clustering model without propagation. Three lines in Figure 2.3 to Figure 2.5 indicate: (1) results with propagation (solid line); (2) results without propagation (dashed line); and (3) results using Latent Semantic Analysis(LSA)-based method by exploiting common words between languages (dotted line). For each figure, 6 plots are taken starting from 0 in 360-point-increments. We conducted the experiments three times for each proportion of supervisory information and use the average values. As we can see from Figure 2.3, Figure 2.4, and Figure 2.5, the propagation method can significantly improve the performance of spectral clustering algorithm. For 1800 documents in 6 topics, we manage to achieve RI $= 0.91$, purity $= 0.84$, and F_2-measure $= 0.76$ with only 20% of documents (360 documents) used as supervisory information. Spectral clustering algorithm without propagation can only achieve $0.69, 0.30, 0.28$ for RI, purity, and F_2-measure respectively. The propagation method is highly effective when only small amount of supervisory information given to the algorithm. Obviously, the more supervisory information given, the better the performance is. As the number of supervisory information increases, the difference of the model performance with and without propagation becomes smaller. This is because there are already enough links

between multilingual documents, so we do not necessarily build more links through similarity propagation anymore. However, even when there are already many links, our model with propagation still outperforms the model without propagation.

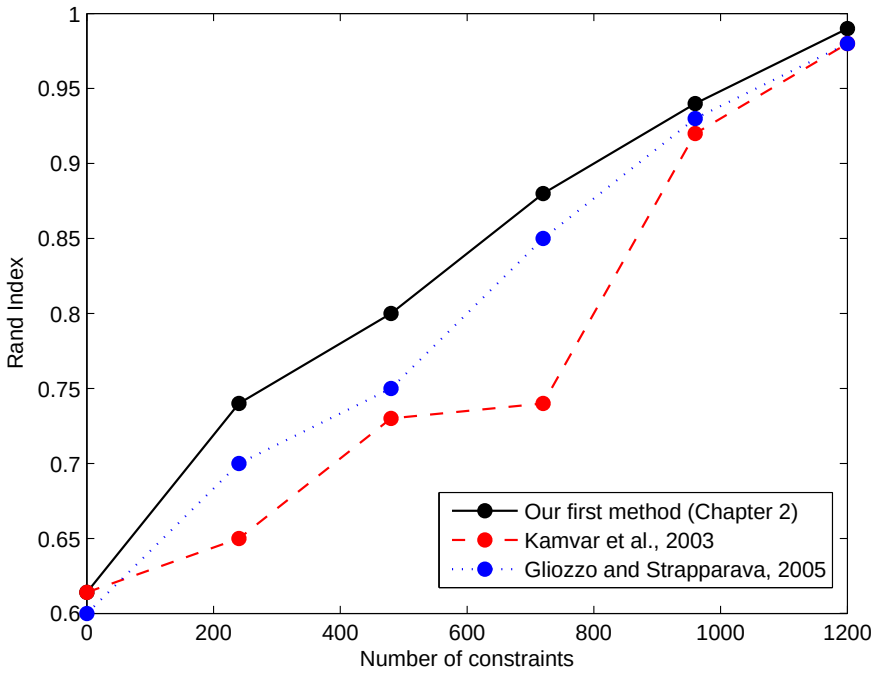
We also compare the performance of our algorithm to LSA-based multilingual document clustering model. We performed LSA to the multilingual term by document matrix. We do not use parallel texts and only rely on common words across languages as well as must-link constraints to build multilingual space. The results show that exploiting common words between languages alone is not enough to build a good multilingual semantic space, justifying the usage of supervisory information in multilingual document clustering task. When supervisory information is introduced, our method achieves better results than LSA-based method. In general, the LSA-based method performs better than the model without propagation.

We assess the sensitivity of our algorithm to parameter β, the penalty for similarity propagation. We assess the sensitivity of our algorithm to parameter β, the penalty for similarity propagation. We tested our algorithm using various β, starting from 0 to 1 in 0.2-point-increments, while other parameters being held constant. Figure 2.6(a) shows that changing β to some extent affects the performance of the algorithm. However, after some value of reasonable β is found, increasing β does not have significant impact on the performance of the algorithm. We also tested our algorithm using various k, starting from 0 to 100 in 20-point-increments. Figure 2.6(b) reveals that the performances of the model with different k are comparable, as long as k is not too small. However, using too large k will slightly decrease the performance of the model. Too many propagations make several dissimilar documents receive high similarity value that cannot be nullified by the post-processing step. Last, we experimented using various t ranging from 2 to 20. Figure 2.6(c) shows that the method performs best when $t = 10$, and for reasonable value of t the method achieves comparable performance.

To show the effectiveness of our clustering algorithm in handling multilingual documents in different writing systems such as English and Japanese, we performed the same experiment on our second dataset. We took approximately 20% of documents in the dataset as supervisory information and created constraints between them. Since the LSA-based algorithm cannot be applied to documents in different writing systems, we only compare our algorithm with the baseline method, i.e., the semi-supervised spectral clustering algorithm without similarity propagation. As we can see from the results in Figure 2.7, Figure 2.8, and Figure 2.9, our method is able to discover good multilingual clusters for documents in different writing systems. The method is particularly superior when only small number of supervisory information is given to the algorithm.

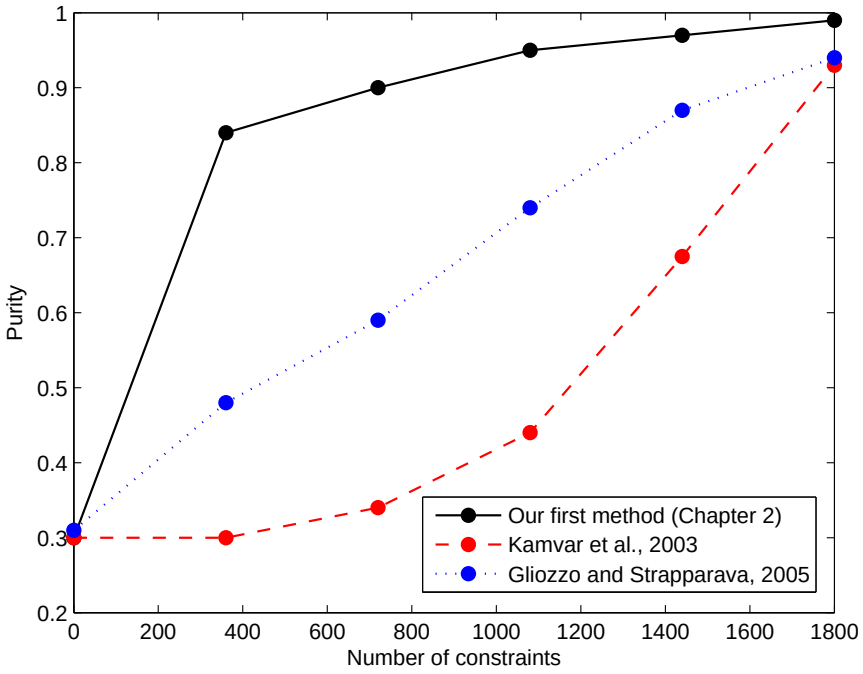


(a) Rand Index for 6 topics

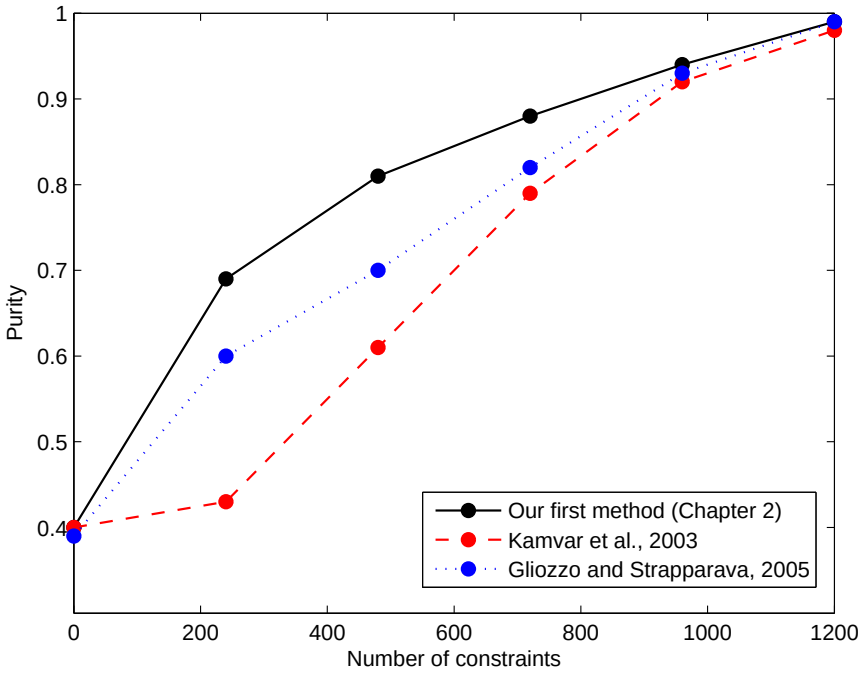


(b) Rand Index for 4 topics

Fig. 2.3. Rand Index on the first dataset with (a) 1800 documents, 6 topics; and (b) 1200 documents, 4 topics as the number of constraints increases. $k = 30, \delta = 0.03, \beta = 0.5, t =$ number of topics, and $S = 2$.

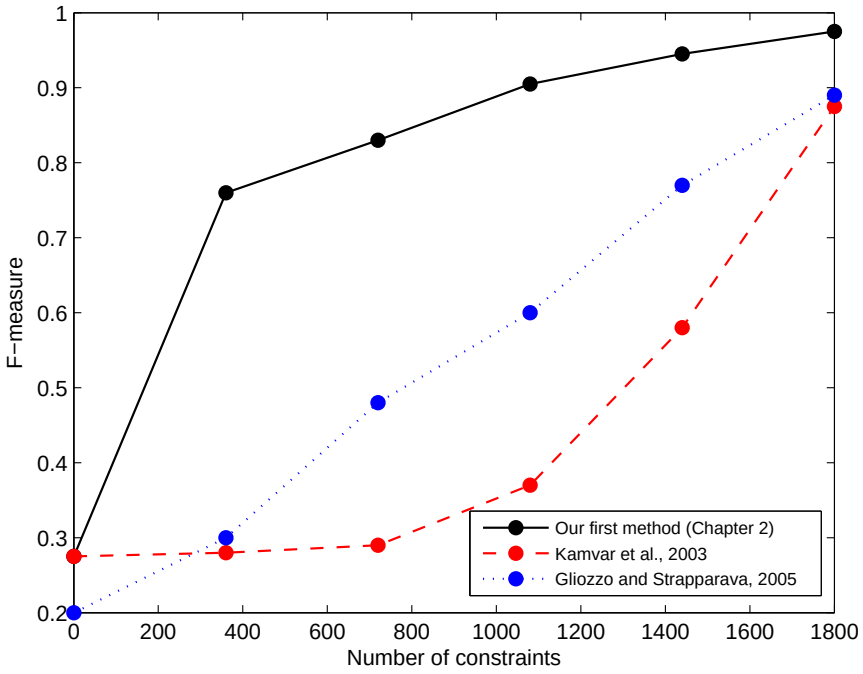


(a) Purity for 6 topics

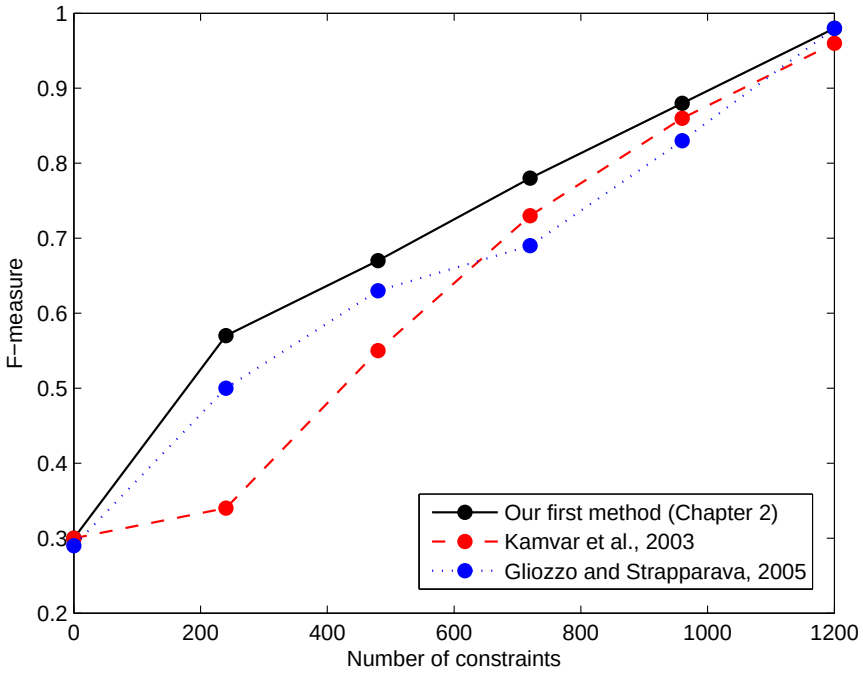


(b) Purity for 4 topics

Fig. 2.4. Purity on the first dataset with (a) 1800 documents, 6 topics; and (b) 1200 documents, 4 topics as the number of constraints increases. $k = 30, \delta = 0.03, \beta = 0.5, t =$ number of topics, and $S = 2$.

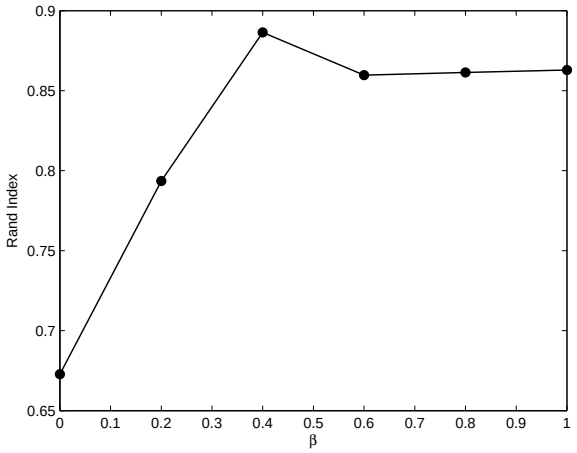


(a) F_2-measure for 6 topics

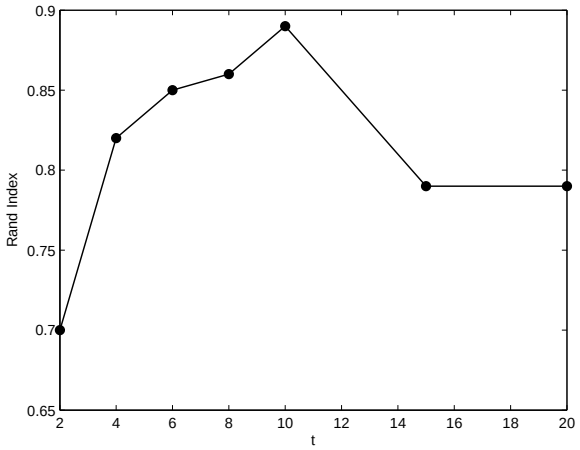


(b) F_2-measure for 4 topics

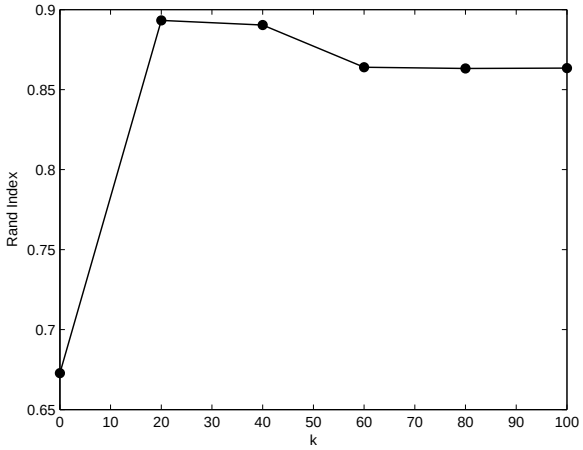
Fig. 2.5. F_2-measure on the first dataset with (a) 1800 documents, 6 topics; and (b) 1200 documents, 4 topics as the number of constraints increases. $k = 30, \delta = 0.03, \beta = 0.5, t =$ number of topics, and $S = 2$.



(a) Changing $\beta, k = 30, t = 6$



(b) Changing $k, \beta = 0.5, t = 6$



(c) Changing $t, \beta = 0.5, k = 30$

Fig. 2.6. Rand Index on the first dataset with 1800 documents and 6 topics as (a) β increases; (b) k increases; and (c) t increases. $\delta = 0.03, S = 2$, and 20% of documents are used as supervisory information.

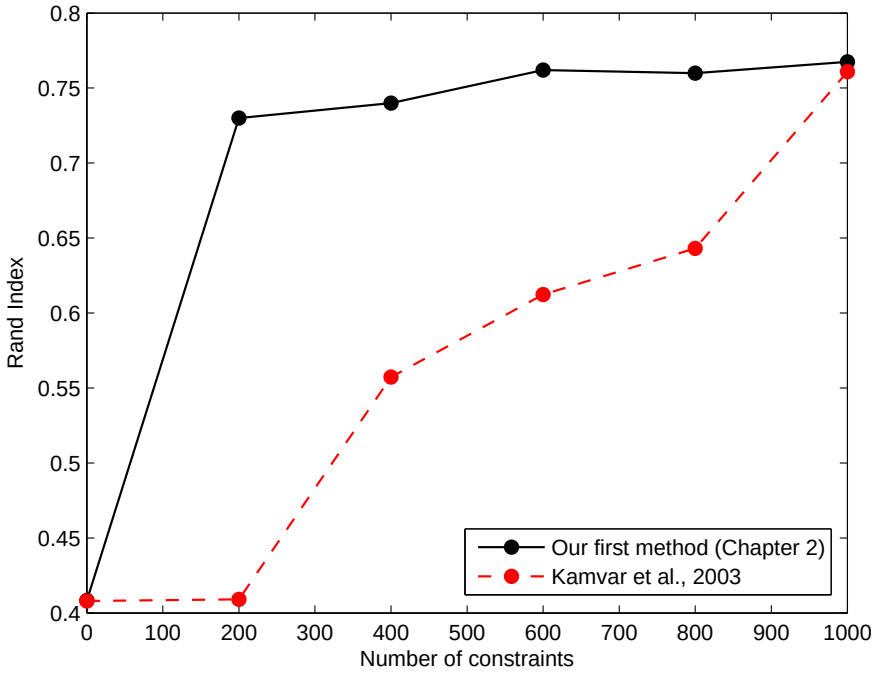


Fig. 2.7. Rand Index on the second dataset with 2279 documents and 3 topics as the number of constraints increases. $k = 20, \delta = 0.05, \beta = 0.5, t = 3$, and $S = 2$.

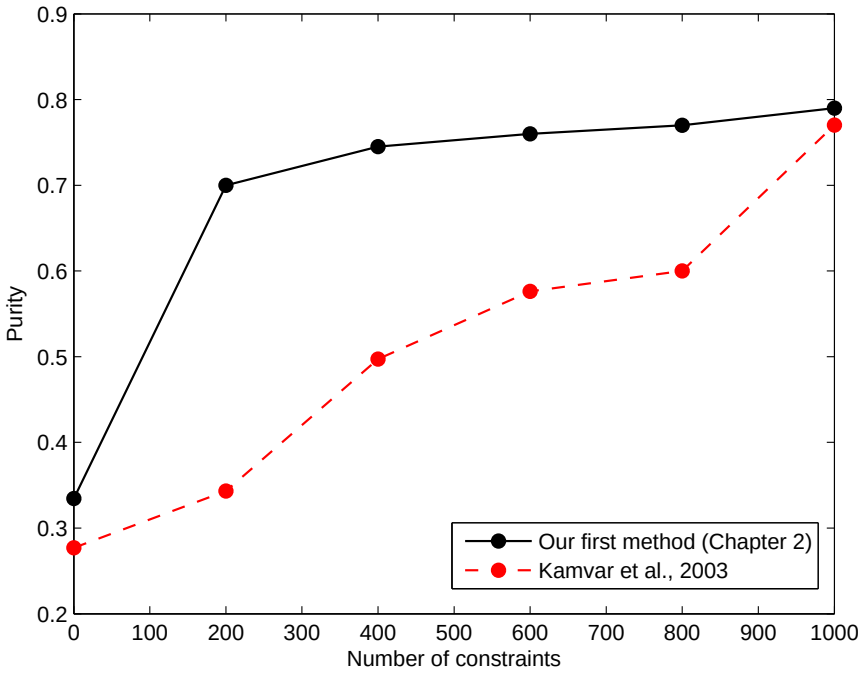


Fig. 2.8. Purity on the second dataset with 2279 documents and 3 topics as the number of constraints increases. $k = 20, \delta = 0.05, \beta = 0.5, t = 3$, and $S = 2$.

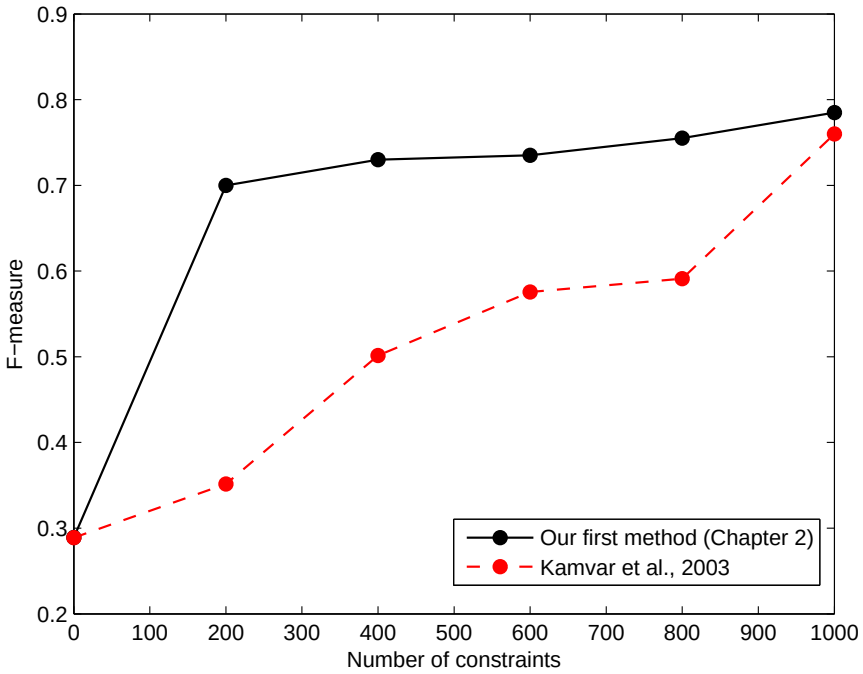


Fig. 2.9. F_2-measure on the second dataset with 2279 documents and 3 topics as the number of constraints increases. $k = 20, \delta = 0.05, \beta = 0.5, t = 3$, and $S = 2$.

Chapter 3

Multilingual Document Clustering Using Web-Searches

This chapter describes a different approach for multilingual document clustering by using web search engine to compute semantic relatedness between multilingual words and further utilizing it to estimate text-to-text similarity. Unlike previous works in multilingual clustering, including our first method, which require the presence of dictionaries, multilingual thesaurus, parallel corpora, or comparable corpora to work; this method is completely unsupervised since it does not need any kind of supervisory information to be given to the algorithm.

The utilization of search engine to measure semantic relatedness between monolingual words has become prevalent in the field of word similarity. These methods usually combine web-count results with text snippets returned by a Web search engine to construct feature vectors for computing the word similarity score [4, 5, 6]. Our method extends the utilization of search engine to compute similarity score between multilingual words. However, it does not use text snippets information and only estimates the similarity score using web-count results.

The word clustering algorithm used in our method is motivated by the method proposed in [19]. The crucial difference is that their method is designed to work in monolingual context and uses the Newman clustering algorithm [23] to partition a graph of words into clusters, whereas our algorithm uses a variant of hierarchical agglomerative clustering algorithm to discover the word clusters.

Our major contribution here is twofold. We present a multilingual word clustering algorithm, and further use it as an intermediate step to estimate semantic relatedness between multilingual documents.

3.1 Problem definition

We formulate the problem of multilingual document clustering as estimating semantic relatedness between texts by computing similarity scores of words contained in them. In order to do this, we first extract informative words in each language involved (Algorithm 3, line 1-3) and query a search engine using each pair of extracted terms as keywords to construct a web-count based word similarity matrix (Algorithm 3, line 4-6). We then use the matrix to guide a hiearchical agglomerative clustering algorithm to find multilingual word clusters (Algorithm 3, line 7). Lastly, the resulting word clusters are utilized as features to perform document clustering (Algorithm 3, line 8-9).

In summary, our document clustering algorithm consists of four steps : term extraction, computation of similarity scores of multilingual words, word clustering, and document clustering. The complete algorithm for unsupervised multilingual document clustering is presented in Algorithm 3.

Algorithm 3 Unsupervised multilingual document clustering

Input: Multilingual documents
Output: Document clusters

1: **for all** language $\lambda \in L$ **do**
2: Select informative terms in λ using Eq. 3.1.
3: **end for**
4: **for all** pair of extracted informative terms **do**
5: Query search engine to compute its similarity score (Eq. 3.2)
6: **end for**
7: Construct word clusters (Algorithm 4)
8: Map document collection to feature space by using word clusters as features
9: Perform k-means clustering in the feature space
10: Output document clusters

3.2 Term Extraction

The first step of our algorithm is analogous to feature selection in a standard document clustering algorithm, which is done to reduce the number of dimensions. However, in order to avoid confusion with the features of our clustering algorithm, which are not the set of these extracted words but clusters of them, we use term extraction to denote this step. The goal of this step is to select a set of informative terms from the document collection. The selection is performed since we need to query search engine using each pair of words to estimate its similarity score, and the number of search that is needed to be done is $C(|W|, 2) + |W|$ times, where W is the set of words in all languages, and $|W| = |W_1| + |W_2| + |W_3| + \ldots + |W_L|$, for L languages in the document collection. Since W is huge, we must select its subset and only use the subset when querying a search engine. The selection of informative words is of paramount importance to the success of our clustering algorithm. We use term variance quality [9] to select informative words from the set of documents in each language. Term variance quality of a term t is defined as :

$$q_0(t) = \sum_{j=1}^{D} f_j^2 - \frac{1}{|D|}\left[\sum_{j=1}^{|D|} f_j\right]^2, \tag{3.1}$$

where f_j is the frequency of term t in document d_j, and $|D|$ is the total number of documents in the collection. We compute $q_0(t)$ for all terms in W and extract top E terms in each language. We use these extracted terms to construct term-similarity matrix as described in the following section.

3.3 Web-count Based Multilingual Word Similarity

Our method uses search engine to estimate similarity score between word A and word B. Web-count for the query *"A" AND "B"* can be considered as an approximation of co-occurrence of A and B on the Web. A web-count based similarity measure also considers the number of occurrences of word A and B alone when computing the similarity score to get a more accurate assessment of semantic similarity between word A and B.

There are several web-count based methods for computing word similarity from search engine results. Examples of which include : Jaccard, Dice, Overlap (Simpson), and Pointwise Mutual Information. In our algorithm, we use Dice coefficient to compute similarity score between mul-

tilingual words. Following [4] and denoting the web-count of query of word A as $H(A)$ and the web-count of query *"A" AND "B"* as $H(A \cap B)$, we define the Dice coefficient as

$$\text{Dice(A,B)} = \begin{cases} 0 & \text{if}(H(A \cap B) \leq S) \\ \frac{2H(A \cap B)}{H(A)+H(B)} & \text{otherwise} \end{cases} \tag{3.2}$$

We argue that the web-count based similarity score for semantically related multilingual words is higher than that of unrelated ones. Therefore, we can use the information retrieved by the search engine to assess semantic similarity between multilingual words. For example[*1], the Dice score of the pair `baseball` - 野球 (baseball) is Dice(`baseball`, 野球)= 0.033, whereas Dice(`science`, 野球)= 0.002. When the words are not translations of each other but semantically related to some extend, such as in the case of `soccer` and 野球, its Dice coefficient Dice(`soccer`, 野球)= 0.014 is still higher than that of a pair of semantically unrelated multilingual words (e.g., `science` - 野球).

Therefore, in this step, for each combination of pair of words in the set of extracted terms, we query a search engine using keywords "A", "B", and *"A" AND "B"*. We then take the resulting web-count to compute its Dice coefficient, and construct a word similarity matrix which is used in the next step to discover multilingual word clusters.

3.4 Multilingual Word Clustering

Given a set of multilingual words and their similarity matrix from the previous step, Algorithm 4 describes the clustering process of these words.

For each word in the set, we choose N most similar words in every other language (Algorithm 4, line 1-3). Next, we define a threshold T of occurrences and remove the words that are selected as top-N by $\geq T$ words in the parent language (Algorithm 4, line 4-10). For example, supposed that we want to cluster English and Japanese documents, and we have a set of extracted Japanese terms { 研究 (research), 野球 (baseball), 科学 (science), 物理 (physics)}, and a set of extracted English terms {`professor, research, science, computer, Japan, soccer, university`}. We also define $N = 3$ and $T = 3$ and use $\delta(j, N)$ to denote N most similar English words to a particular Japanese word j. Supposed that $\delta($ 研究$, 3) = \{$*Japan, research, computer*$\}$, $\delta($ 野球$, 3) = \{$`Japan, university, soccer`$\}$, $\delta($ 科学$, 3) = \{$`Japan, science, professor`$\}$, and $\delta($ 物理$, 3) = \{$`Japan, computer, university`$\}$. The pre-processing step of the word clustering algorithm will remove the word `Japan` since it appears four times in the example, while the threshold T is set to three.

We observe that the removal of these words can reduce the noise in the set of extracted terms. There are English words that co-occur frequently with Japanese words on the Web, such as {`top, home, English`}. It is straightforward to see that these words have high similarity scores due to the way a Web document is presented. A Japanese document on the Web usually contains a link to the top page, the home page, and the English version of the document. Therefore, we will get many co-occurrences of these English words with almost all Japanese words, while actually they are not semantically related. Other possible source of this noise includes words which are highly related to the language, independent of the concepts, such as the word `Japan` in our example. Since we compare Japanese and English words, naturally the word `Japan` occurs frequently with almost every word and thus always has high similarity scores. The removal process is analogous to inverse document frequency in term weighting problem, where we assign higher weight to an infrequent term. However, in our method, to reduce the complexity of the step, we just nullify the weight of words which occur more than T times. Note that as a result of this step, each word does not necessarily have the same number of most similar words.

[*1] We use Google search engine (http://www.google.com) in this example.

The result of the previous step is used as the initial word clusters of our complete-link hierarchical agglomerative clustering (HAC) algorithm. Note that a particular initial word clusters consists of a single word in a parent language, along with its most similar words in other language(s). We divide our initial word clusters based on the parent language to get $|L|$ sets of initial word clusters, where $|L|$ is the number of languages in the collection. Using the example above, our initial word clusters after T thresholding for Japanese as the parent language are $\overline{\delta}$(研究, 3) = { 研究 (research) , research, computer}, $\overline{\delta}$(野球, 3) ={ 野球 (baseball) , university, soccer}, $\overline{\delta}$(科学, 3) ={ 科学 (science), science, professor}, $\overline{\delta}$(物理, 3) ={ 物理 (physics) , university, computer}. We will also have other initial word clusters in which English is the parent language : $\overline{\delta}(\texttt{professor}, 3), \overline{\delta}(\texttt{research}, 3), \ldots, \overline{\delta}(\texttt{science}, 3)$.

Denoting the word in the parent language for an initial cluster X as X_P and the words in language λ as $X_{\{O_\lambda\}}$, we compute the similarity between two initial clusters Sim(X,Y) as

$$\frac{\text{Sim}(X_P, Y_P) + \sum_{\lambda \in L, \lambda \neq P} \text{Sim}(X_{\{O_\lambda\}}, Y_{\{O_\lambda\}})}{|L|} \tag{3.3}$$

where Sim(X_P, Y_P) is the similarity score between word X_P and Y_P; and Sim($X_{\{O_\lambda\}}$, $Y_{\{O_\lambda\}}$) is

$$\frac{\sum_{x \in X_{\{O_\lambda\}}} \sum_{y \in Y_{\{O_\lambda\}}} \text{Sim}(x,y)}{|X_{\{O_\lambda\}}| + |Y_{\{O_\lambda\}}|} \tag{3.4}$$

Note that in the computation of initial word clusters similarity, we average the similarities of words that are in the *same* language. If one or both of the clusters do not have similar words in language λ, we set Sim($X_{\{O_\lambda\}}$, $Y_{\{O_\lambda\}}$) to zero when computing the similarity score between X and Y, and subtract the denominator in Equation 3.3 by one for each λ that does not have its match in the other cluster. Figure 3.1 gives an illustration of the steps described above.

We merge this initial clusters using a complete-link HAC algorithm (Algorithm 4, line 13). As a complete-link HAC algorithm, at each clustering step, the algorithm joins two most similar clusters X and Y and represent the new similarity scores with other clusters as the minimum of their similarities with cluster X and cluster Y. We stop the merging process when the similarity score of the most similar clusters is below a predefined threshold H. Using the priority queue implementation in [18], the complexity of the HAC algorithm is $\Theta(N^2 \log N)$.

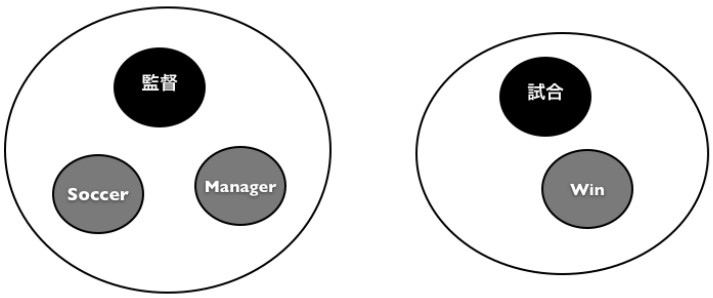
Since we divide our initial word clusters to $|L|$ distinct sets based on their parent language, we perform the HAC algorithm $|L|$ times, where $|L|$ is the number of languages in the collection (Algorithm 4, line 12-15). We combine all results of the word clustering algorithm and use them as features as described in the following section.

3.5 Unsupervised Document Clustering

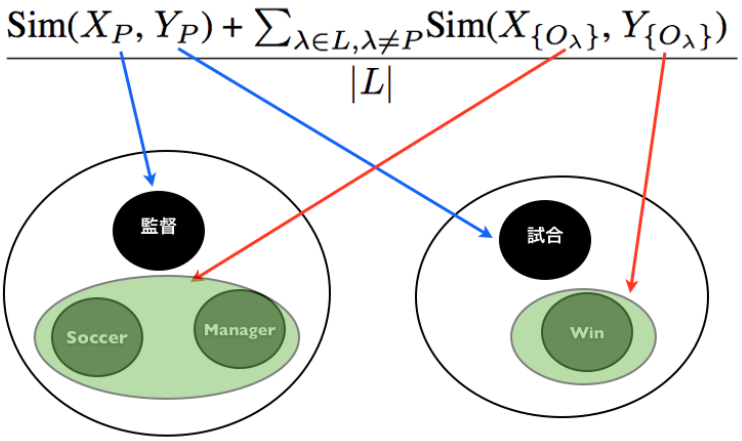
We use the resulting clusters of multilingual words Γ from the previous step as features by considering a cluster as a feature. Therefore, the number of dimension of a document vector in the feature space is equal to the total number of word clusters in $|L|$ runs of the word clustering algorithm. Alternatively, we can remove singletons (initial word clusters which are not merged with any other cluster during the word clustering process) from the results of word clusters and use the remaining clusters as features.

Continuing from the previous example, supposed that after word clustering and removal of singletons, the resulting word clusters are Γ_1 = { 研究 (research), 科学 (science), 物理 (physics), science, research, professor, computer}, Γ_2 = { 野球 (baseball), university, soccer} when Japanese is used as the parent language; and Γ_3 = {research, science, university, 研究 (research)} when English is used as the

(a) Suppose that we extract two Japanese words 監督 and 試合



(b) Find similar words in the set of extracted terms in other language(s), in this case, English. We use cross-language word similarity score to find most semantically related English words to the Japanese words.



(c) Computation of similarity between two initial word clusters. In this example, the parent language is Japanese and the child language is English. We first average similarity between words in the same language, and then average similarity between languages. Note that cross-language similarity between words is not used in this step.

Fig. 3.1. Illustration of how to transform extracted terms into initial word clusters and compute their similarity.

parent language. The document vector is a three dimensional vector where each dimension corresponds to a cluster. Note that a term might appear in more than one word cluster.

We map the documents to the feature space as follows. For a document $d_j = \{w_1, w_2, \ldots, w_n\} \in D$, if w_i is represented in the word clusters result, f_j denotes the occurrences of w_i in document d_j. We also compute the number of occurrence of w_i in all documents, and define inverse document frequency idf_i as $\log \frac{|D|}{|d_{w_i}|}$, where $|D|$ is the number of documents and $|d_{w_i}|$ is the number of documents containing word w_i. We ignore words which do not occur in the word clusters result. We denote the weight of a particular term w_i in document d_j as $v_{ij} = f_j.idf_i$. The weight of the m-th dimension (which corresponds to the m-th word cluster in Γ) of a document

Algorithm 4 Word clustering algortihm

Input: A set of extracted terms in each language involved Φ, its web-count based similarity matrix M, number of similar words N, occurrence threshold T, clustering threshold H
Output: Word clusters Γ

1: **for all** word w_i in Φ **do**
2: $\delta(w_i, N) \leftarrow N$ most similar words in every other language
3: **end for**
4: **for all** word w_i in Φ **do**
5: Count(w_i) $\leftarrow$ number of times w_i is selected as N most similar words
6: **if** Count(w_i) $\geq T$ **then**
7: Remove all occurrences of w_i in δ
8: Remove w_i from Φ
9: **end if**
10: **end for**
11: $\Gamma \leftarrow []$
12: **for all** language $L_i \in L$ **do**
13: word-clusters$_{L_i}$ $\leftarrow$ HAC(Φ_{L_i}, H) {Perform hierarchical agglomerative clustering algorithm on Φ_{L_i}, use Eq. 3.3 and Eq. 3.4 to compute initial clusters similarity, stop when the clustering threshold is lower than H. See [18] for the implementation of HAC algorithm}
14: Γ.APPEND (word-clusters$_{L_i}$)
15: **end for**
16: Output the word clusters Γ.

d_j in the feature space is computed as

$$\sum_{w_c \in \Gamma_m} \frac{v_{cj}}{|\eta_c|}, \tag{3.5}$$

where Γ_m is the m-th cluster in Γ, η_c is the number of words in the same language as w_c in cluster Γ_m, and v_{cj} is the weight of term w_c in document d_j.

Last, we normalize the document vector by its length to create a unit vector, before performing k-means clustering algorithm in the feature space to discover multilingual document clusters.

3.6 Experiments

3.6.1 Data set

We used a subset of Japanese and English documents from Reuters RCV-2 Corpus. The documents in this corpus are not parallel, but are written at the same period by local reporters in each language. We used documents in three topics : weather (GWEA), war and violence (GVIO), and economy (E212). Table 3.1 shows the number of documents used in our experiments.

	War	**Weather**	**Economy**	**Total**
Japanese	235	468	390	1093
English	392	468	326	1186
Total	627	936	716	2279

Table. 3.1. Number of documents in our experiments.

English	*prisoners explosion clear eurobond witness federal anticipation warm daily currency arrests political written yen noon blasted crowns supply rainfall produce typhoon deposit president sudanese shell strikes cloudy military morning skies hizbollah economists managers*
Japanese	ドル (dollar) - 入札 (bid) - 国債 (national debt) - 予報 (forecast) - 降雨 (rainfall) - 人質 (hostage) - 外貨 (foreign currency) - 被害 (damage) - カ氏 (Fahrenheit) - 気温 (temperature) - セ氏 (Celsius) - 気象 (weather) - 王国 (kingdom) - 穀物 (grain) - ユーロ (Euro) - 予想 (forecast) - 利率 (interest rate) - 乾燥 (dry) - 償還 (repayment) - 小麦 (wheat) - 爆発 (explosion) - エルニーニョ (El Nino) - 寒気 (cold air) - 大使 (ambassador) - 生産 (production) - 強風 (strong wind) - イラク (Iraq)

Table. 3.2. Subset of extracted terms.

3.6.2 Evaluation methods

Normalized mutual information (NMI) was used to analyze the clustering results. It measures the amount of statistical information shared by the random variables representing the cluster assignments (output of the algorithm) and the true assignments of the data. NMI is defined as

$$NMI(l, L) = \frac{I(l; L)}{[H(l) + H(L)]/2}, \tag{3.6}$$

where l is the output assignments of the algorithm, L is the true assignments, $I(l; L)$ is the mutual information, and H is the Shannon entropy.

We also evaluated the purity of the resulting document clusters. We treated a cluster as a class by assigning it to the most frequent topic of its members. We computed the accuracy as the fraction of documents that are correctly classified by the algorithm.

3.6.3 Pre-processing

For English documents, we removed stopwords and performed Porter stemming algorithm to reduce the number of words. However, since we needed the word to be in meaningful unit when querying the search engine[*2], we used the most frequent form to label a conflation class. For example, the words `determine, determined, determinative, determinable, determinate` are stemmed to the same conflation class, `determin`. We considered all these words as a single word, but labeled the conflation class as `determine`, since it is the most frequent form in our corpus.

For Japanese documents, we parsed the original documents using MeCab [16]. We removed single character hiragana and katakana and used part-of-speech information from MeCab output to reduce the number of original words by discarding all conjunctions and modals.

After pre-processing, we were left with 11847 English words and 3911 Japanese words, for a total of 15758 words.

3.7 Results and discussions

We defined the parameters of our algorithm as follows : $E = 500, S = 0.00005, N = 10, T = 20, H = 0.4$. E is the number of selected terms in the term extraction step, S is the minimum similarity score between two words (otherwise zero), N is the number of similar terms selected, T is

[*2] We use Bing search engine (http://www.bing.com) in our experiments.

the occurrence threshold of a term when constructing initial word clusters, and H is the similarity score used as the stop condition of the hierarchical agglomerative word clustering algorithm. We set the threshold S to a small value since we intend to utilize it to discard low similarity score between multilingual words. This nullification is important in order to make sure that a word which does not have N related words in the set of extracted terms in other language, is not forced to pick unrelated word when selecting its N most similar words. By zeroing out the similarity score below S, we ensured that all selected words are closely related to that particular word. Note that if there are less than N related words, we did not require N to be satisfied, since initial clusters of our word clustering algorithm can have different numbers of words.

Table 3.2 shows a subset of words selected by the term extraction step of our algorithm. We can see that the method is able to extract words that are representative of the topics we used. For example, the words {`prisoners, explosion, blasted, hizbollah, striker, military`}, {`warm, daily, rainfall, typhoon, cloudy`}, {`eurobond, currency, economists, managers, deposit`} represent the *war*, *weather*, and *economy* topics respectively. Similarly, for Japanese documents, the table reveals that we have words representing the war, weather, and economy topics.

Table 3.3 shows the similarity score of several words extracted by our algorithm. We can see that semantically related word generally has higher similarity score compared to unrelated one in the same language. For example, 降雨 (rainfall) and 気象 (weather) has a Dice score of 0.0600, while the Dice score for 降雨 (rainfall) and ユーロ (Euro) is 0.0078. Likewise, Dice (降雨, `weather`) = 0.0003 is higher than Dice (降雨, `bonds`) = 0, since 降雨 (rainfall) is more related to `weather` than to `bonds`. However, when we compare a Japanese - Japanese pair of words to Japanese - English pair, the similarity score does not provide a reliable indication of semantic relatedness since Japanese-Japanese pair, even unrelated one, almost always has higher similarity score than Japanese - English pair. Our method deals with this problem by performing language alignment when computing the similarity score between initial word clusters (Equation 3.3 and Equation 3.4). The table also shows that the pair 降雨 - *Japan* has high similarity score, due to the familiarity of the word *Japan* with almost all Japanese words, as described in the previous section. We expect the occurrence-thresholding step before the word clustering process to eliminate this noise. Remember that threshold T is set to 20 in our experiment.

Table 3.4 shows several examples of word clusters discovered by our algorithm. We can see

Words pair	**Dice**	**Related**
降雨 - 気象 (weather)	0.0600	YES
降雨 - 気温 (temperature)	0.0410	YES
降雨 - 人質 (hostage)	0.0092	NO
降雨 - 入札 (bid)	0.0099	NO
降雨 - ユーロ (euro)	0.0078	NO
降雨 - *weather*	0.0003	YES
降雨 - *winds*	0.0002	YES
降雨 - *flood*	0.0003	YES
降雨 - *bonds*	0.0000	NO
降雨 - *refugee*	0.0000	NO
降雨 - *iraq*	0.0000	NO
降雨 - *tax*	0.0000	NO
降雨 - *Japan*	0.0003	NO

Table. 3.3. Similarity scores with the word 降雨 (rainfall). We query Bing search engine using the keywords *"A"*, *"B"*, and *"A" AND "B"* for each pair A - B in the first column of the table.

Japanese - English	予報 (forecast) - 観測 (observation) - 港湾 (harbor) - *forecast* 小麦 (wheat) - 大麦 (barley) - 作物 (crops) - *maize - wheat - grain - crops* キンシャサ (Kinshasa) - *tutsi - mobutu - zairean - rwandan - kabila* 降雨 (rainfall) - 流域 (river basin) - 気象 (weather) - *rainfall* 外貨 (foreign exchange) -トレーダー (trader) - *stock*
English - Japanese	*cyclone* - サイクロン (cyclone) - 気圧 (atmospheric pressure) - 天気 (weather) *rainfall* - 気象 (weather) - 降雨 (rainfall) - 洪水 (flood) - 雨量 (rainfall) *hizbollah* - ハマス (Hamas) - 停戦 (ceasefire) *iraqi - saddam - iraq* - イラク (Iraq) *wounded - prisoners - killings - arrests* - 警察 (police) *alberta - saskatchewan - manitoba* - カナダ (Canada)

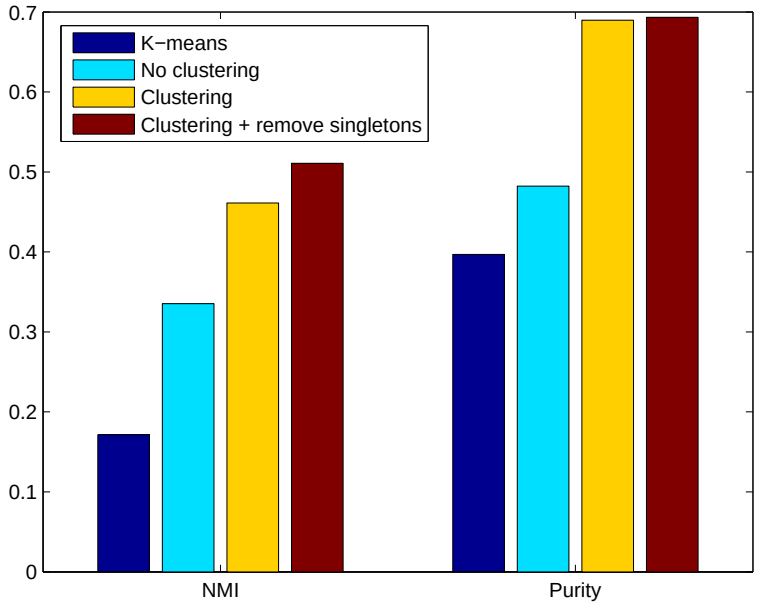
Table. 3.4. Word clusters. The first row (Japanese - English) represents clusters with Japanese as the parent language, while the second row (English - Japanese) represents clusters with English as the parent language. Each row in the right column corresponds to a word cluster. We can see that the word clusters discovered by our method mainly consist of semantically related words.

that each cluster mostly consists of semantically related words, and provides a good basis for predicting the correct cluster of a particular document.

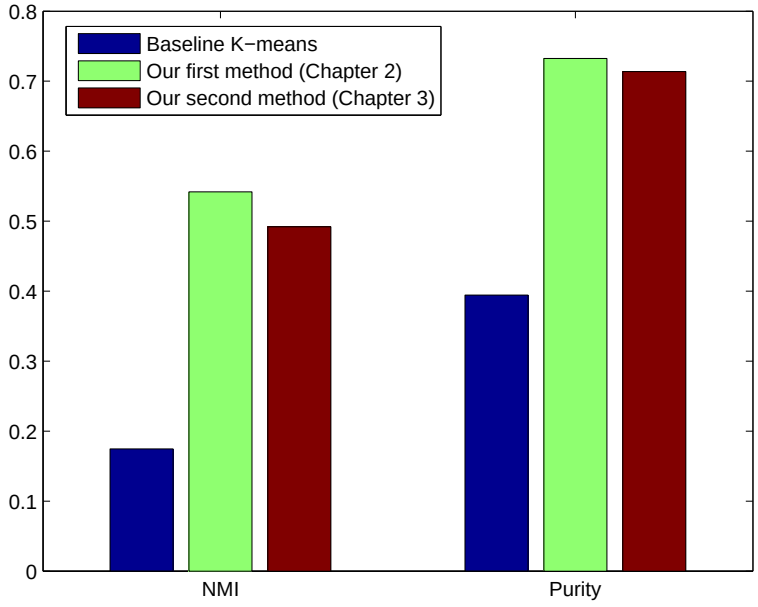
To the best of our knowledge, our method is the first unsupervised method for multilingual document clustering. Therefore, we cannot compare the method with other unsupervised multilingual document clustering algorithm. Moreover, though our dataset is a comparable corpus, it is not parallel and the languages that are used are from different families. For this reason, in our first experiment, we compared our method with its own variants. We provide another comparison using a subset of the data with the semi-supervised clustering algorithm described in the previous chapter in the second experiment. We also used the standard k-means clustering algorithm as the baseline method in both experiments and averaged all results over ten runs of each algorithm. Figure 3.2 shows the performance of each competing algorithm.

In the first experiment, we defined two variants of our document clustering algorithm. The first one is a document clustering algorithm without the word clustering step. The second one performs word clustering but does not remove singletons (clusters with less than three words). Besides clusters with a single word, we considered word clusters consisting of only two words as singletons, since we expected the initial clusters to have at least two words (a word in the parent language and at least one related word in other language). In Figure 3.2(a), the dark blue bar represents the k-means clustering algorithm, and unsurprisingly, the algorithm performs poorly as indicated by very low NMI value and purity. The light blue bar represents the first variant of our multilingual document clustering algorithm, the one without the word clustering step. Therefore, after selecting most similar words, we used all initial clusters as features for multilingual document clustering. The yellow bar represents the second variant of our method (i.e. the one which carries out word clustering without removing singletons afterwards), while the red bar represents the best version of our algorithm which performs word clustering and removal of singletons. We can see that the word clustering step significantly improves the performance of our document clustering model, and the removal of singletons enhances it slightly further. The best performing variant of our algorithm achieves 69% classification accuracy with a NMI value of 0.51.

In the second experiment, we compared our method with the semi-supervised multilingual document clustering presented in Chapter 2. In Figure 3.2(b), the dark blue and red bars represent the same algorithms as in our first experiment (k-means and our best performing algorithm), while the green bar represents the semi-supervised clustering algorithm. Since the semi-supervised approach requires the presence of comparable corpora as supervisory information, we took a subset



(a) First experiment (2279 documents)



(b) Second experiment (1875 documents)

Fig. 3.2. Experimental results for each competing algorithm. The upper bar graph shows the results for the first experiment where we compared three variants of our unsupervised algorithm. The lower bar graph displays the comparison with semi-supervised clustering algorithm presented in the previous chapter. We also provide k-means clustering algorithm as the baseline method in both experiments. In each figure, the y-axis represents NMI value and classification accuracy (purity) as indicated by the label in the x-axis.

of our original collection (≈20%) and used it as supervisory information. We only computed the performance on the test set for all three competing algorithms, but clustered using all documents. The results show that even though our method does not utilize supervisory information, it performs comparably with the semi-supervised clustering algorithm. In terms of accuracy, our unsupervised algorithm classifies 71.3% of the documents correctly, while the semi-supervised algorithm achieves 73.2% classification accuracy. The NMI values of the resulting clusters are 0.49 and 0.54 respectively.

As other methods which use search engine to compute similarity scores between words, a salient weak point of our algorithm is the necessity to query search engine for each pair of words. Querying search engine takes a significant amount of time, which forces us to select only a small subset of original terms to be considered as clustering features. However, our experiments show that using a realistic number of terms selected by the term extraction step (500 in each language), our method is able to achieve promising clustering results.

Perhaps surprisingly, observing the initial clusters of multilingual words after eliminating words occurring above threshold T, we notice that the most similar word to a particular word is often its exact translation (providing that the translation exists in the set of extracted terms). Several examples of which include {*corn*, コーン }, {*coffee*, コーヒー }, {*proposal*, 提案 }, {*hostages*, 人質 }, { 応札, *bid*}, and { セ氏, *Celsius*}. While we do not thoroughly explore the possibility, we believe that our word clustering method can be used as a basis for mining alignment between bilingual words from comparable corpora.

Chapter 4

Conclusion and Future Work

4.1 Conclusion

In this thesis, we present two methods for multilingual document clustering which answer common problems of existing algorithms. Our focus is on estimating text-to-text semantic relatedness of multilingual documents through various methods, either explicitly using comparable corpora or implicitly by evaluating semantic relatedness of words which are contained in the documents.

In chapter 2, we propose a multilingual spectral clustering model which uses comparable corpora as pairwise constraints and performs similarity propagation to estimate multilingual texts semantic relatedness. Since the method takes supervisory information in the form of pairwise constraints, it can be considered as a semi-supervised clustering method. Unlike previous methods using comparable corpora, our method does not try to exploit the presence of common words across languages using co-occurrences based method such as Latent Semantic Analysis, which make them unable to be generalized to collection of documents in languages of different writing systems. Specifically, our method models collection of multilingual documents as undirected weighted graph and similarity between monolingual documents is computed using cosine similarity function of document vectors. Since almost all the time the cosine similarity between multilingual documents is zero, based on the idea of text-to-text semantic relatedness of documents in comparable corpora, we devised a method called similarity propagation to estimate this value. The last step of the algorithm finds the best partition of the propagated graph using spectral clustering algorithm to discover document clusters.

Chapter 3 introduces an ”unsupervised” model which does not require the presence of typical supervisory information such as dictionary, thesaurus, parallel texts, and comparable corpora when clustering multilingual documents. The method uses search engine to compute word similarity and utilizes the word clusters to estimate semantic relatedness of multilingual texts. Specifically, it consists of four main steps, namely term extraction, web searches, word clustering, and document clustering. In the term extraction step, we select informative words from document collection by ranking them using term variance quality [9]. For each combination of pair of words in the set of extracted terms, we query search engine to compute their similarity using web-count based similarity score (Dice coefficient) and build a word similarity matrix. This matrix is utilized in the third step where we cluster semantically related words to create language independent features, which are then used in the last step to cluster multilingual documents.

We show that using only limited supervisory information, the first algorithm significantly outperforms existing comparable-corpora based method. Since comparable corpora are easy to be acquired, the method has the potential to be adapted to documents in various languages, including minor languages in which parallel texts are still scarce. The characteristics of the algorithm also allow it to be applied to languages in different writing systems, as proven by experiments using English and Japanese documents. The second algorithm is the first unsupervised method proposed in multilingual document clustering. We show that its performance is comparable to

the state-of-the-art semi-supervised method (our first approach). Similar to the first method, our experiments also demonstrate that the method is applicable to collection of documents written in different writing systems.

4.2 Future Work

We slightly discuss the possibility of the second algorithm to be used to mine alignment between multilingual words from comparable corpora in Chapter 3. In the future, we will explore this possibility and try to develop an alignment-miner algorithm based on our word clustering method. We believe that the utilization of search engine can help deciding the correct alignment of multilingual words. A successful implementation has the potential to boost the performance of a statistical machine translation system, as well as for creating multilingual lexicons from comparable corpora. There is, however, a typical drawback of web searches-based method (i.e., querying search engine is slow) that has to be solved for widespread application of the method in various domains.

We also plan to combine comparable corpora and web-searches to obtain a more robust clustering algorithm and further improve the state-of-the-art in multilingual document clustering. The simplest solution is to average the similarity values from both methods. However, more advance solution which integrates the strength of each method need to be designed to achieve an even better clustering performance.

Since web searches takes significant amount of time, other interesting possibility is to devise a method that is analogous to our similarity propagation for multilingual words. The goal is to replace the web searches step for estimating similarity between multilingual words. One possible idea is to treat all documents in comparable corpora in the same topic as one group and estimate the similarity based on the number of co-occurrences of those words in all groups. Another feasible idea, if we have access to huge comparable corpora, is to eliminate the web searches and use the comparable corpora to compute the number of co-occurrences of each pair of word. However, the new method will not be unsupervised anymore since it requires comparable corpora as supervisory information. Other interesting idea includes using comparable corpora if there are adequate evidences in them and resorting to web searches when the information is not sufficient. Nonetheless, the similarity values might not be comparable, so a new method has to be devised in order to use them interchangeably.

While it is pretty straightforward to adapt our methods for multilingual text categorization problem, there are still open questions as to whether the method can be extended to other related problems such as cross-language text categorization and cross-language information retrieval. Cross-language text categorization requires training a classifier in one language to be used for classifying documents in other language. On the other hand, cross-language information retrieval system has to retrieve documents in language different than that of the query. A notable difference is that a query is significantly shorter than a document. However, we think that it is possible to compute the similarity between a query and a set of documents in the same language as the query, and propagate the similarity to documents to be retrieved using comparable corpora. We believe that we can also use semantically related word clusters as features in cross-language information retrieval to estimate similarity between query and documents, as in our second approach. We can do the same for cross-language text categorization by replacing query with the document to be classified. It is interesting to see how similarity propagation and multilingual word clusters actually perform in these domains.

Publications and Research Activities

(1) Dani Yogatama and Kumiko Tanaka-Ishii. *Multilingual Spectral Clustering Using Document Similarity Propagation*. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 871–879.

References

[1] Charu C. Aggarwal, Stephen C. Gates, and Philip S. Yu. On the merits of building categorization systems by supervised clustering. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 352–356, 1999.

[2] Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. A study on similarity and relatedness using distributional and wordnet-based approaches. In *NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[3] Satanjeev Banerjee and Ted Pedersen. Extended gloss overlaps as a measure of semantic relatedness. In *IJCAI'03: Proceedings of the 18th international joint conference on Artificial intelligence*, pages 805–810, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[4] Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka. Measuring semantic similarity between words using web search engines. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 757–766, 2007.

[5] Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka. A relational model of semantic similarity between words using automatically extracted lexical pattern clusters from the Web. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 803–812, August 2009.

[6] Hsin-Hsi Chen, Ming-Shun Lin, and Yu-Chuan Wei. Novel association measures using web search with double checking. In *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 1009–1016, 2006.

[7] D. Cutting, D. Karger, J. Pedersen, and J. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1992.

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

[9] I. S. Dhillon, J. Kogan, and M. Nicholas. *A Comprehensive Survey of Text Mining*, chapter 4. Feature Selection and Document Clustering, pages 73–100. Springer-Verlag, 2003.

[10] Susan T. Dumais, T. A. Letsche, M. L. Littman, and T. K. Landauer. Automatic cross-language retrieval using latent semantic indexing. In *In AAAI Spring Symposium on Cross-Language Text and Speech Retrieval*, 1997.

[11] D.K. Evans and J.L. Klavans. A platform for multilingual news summarization. In *Technical Report, Department of Computer Science, Columbia University*, 2003.

[12] Miroslav Fiedler. A property of eigenvectors of nonnegative symmetric matrices and its applica- tions to graph theory. *Czechoslovak Mathematical Journal*, 25:619–672, 1975.

[13] Alfio Gliozzo and Carlo Strapparava. Cross language text categorization by acquiring multilingual domain models from comparable corpora. In *ParaText '05: Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 9–16, 2005.

[14] Sepandar D. Kamvar, Dan Klein, and Christopher D. Manning. Spectral learning. In *In*

IJCAI, pages 561–566, 2003.

[15] Dan Klein, Sepandar D. Kamvar, and Christopher D. Manning. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning*, pages 307–314, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[16] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 230–237, July 2004.

[17] Xiaoyong Liu and W. Bruce Croft. Cluster-based retrieval using language models. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 186–193, 2004.

[18] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition, July 2008.

[19] Yutaka Matsuo, Takeshi Sakaki, Kôki Uchiyama, and Mitsuru Ishizuka. Graph-based word clustering using a web search engine. In *EMNLP '06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 542–550, 2006.

[20] Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Judith L. Klavans, Ani Nenkova, Carl Sable, Barry Schiffman, and Sergey Sigelman. Tracking and summarizing news on a daily basis with columbia's newsblaster. In *Proceedings of the second international conference on Human Language Technology Research*, pages 280–285, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[21] Marina Meila and Jianbo Shi. Learning segmentation by random walks. In *In Advances in Neural Information Processing Systems*, pages 873–879. MIT Press, 2000.

[22] Marina Meila and Jianbo Shi. A random walks view of spectral segmentation. In *AI and STATISTICS (AISTATS) 2001*, 2001.

[23] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69:066133, 2004.

[24] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[25] Bruno Pouliquen, Ralf Steinberger, Camelia Ignat, Emilia Käsper, and Irina Temnikova. Multilingual and cross-lingual news topic tracking. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 959, 2004.

[26] Gerard Salton. Cluster search strategies and the optimization of retrieval effectiveness. In *In The SMART Retrieval System - Experiments in Automatic Document Processing*, pages 223–242, 1971.

[27] Stefan Siersdorfer and Sergej Sizov. Restrictive clustering and metaclustering for self-organizing document collections. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 226–233, New York, NY, USA, 2004. ACM.

[28] Kiri Wagstaff and Claire Cardie. Clustering with instance-level constraints. In *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learning*, pages 1103–1110, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[29] Chih-Ping Wei, Christopher C. Yang, and Chia-Min Lin. A latent semantic indexing-based approach to multilingual document clustering. *Decis. Support Syst.*, 45(3):606–620, 2008.

[30] Oren Zamir and Oren Etzioni. Web document clustering: a feasibility demonstration. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 46–54, New York, NY, USA, 1998. ACM.

[31] Dell Zhang and Robert Mao. Extracting community structure features for hypertext classification. In *ICDIM*, pages 436–441, 2008.