A Term Frequency Range for Text Representation

David Pérez[1], José Tecuapacho[1], Héctor Jiménez-Salazar[1], Grigori Sidorov[2]

[1]Faculty of Computer Science,
Autonomous University of Puebla,
14 sur y Av. San Claudio, C.U.
Edif. 135, CP 72570, Puebla, Pue., Mexico
{dave_p_g,bicha_tecua}@yahoo.com.mx, hgimenezs@gmail.com
[2]Natural Language and Text Processing Laboratory,
Center for Research in Computer Science,
National Polytechnic Institute
Av. Juan Dios Batiz, s/n, Zacatenco, 07738, Mexico City,
Mexico
sidorov@cic.ipn.mx

Abstract. In this work, terms weighting is explored using the Vector Space Model framework. We rely on the hypothesis that terms with medium frequencies have high semantic content. This idea has been exploited before using the Transition Point approach, but manual selection of a threshold is required. We present a formula for determining these "important" frequencies automatically without any threshold setting. We made experiments for Information Retrieval and Text Extraction tasks using three subcollections of TREC-5. The results show that the suggested weighting scheme is a considerable improvement of the Transition Point approach.

1 Introduction

The Vector Space Model framework is very useful approach to text representation in Natural Language Processing tasks, like text categorization, text clustering or automatic summarization. Even when other models, such as probabilistic or fuzzy, are used, this model is mentioned as a good one [6]. In Vector Space Model, each document is represented as a vector, which indexes are all words (terms) used in a given text collection. Namely, for a text collection $\{D_1, ..., D_M\}$ with terms $V = \{t_1, ..., t_n\}$, the vector $\overrightarrow{D_i}$ of dimension n that corresponds to the document D_i is composed of entries d_{ij} with weight of each term t_j in D_i obtained according to the formula:

$$d_{ij} = tf_{ij} * idf_j, \tag{1}$$

where tf_{ij} is frequency of the term j in the document i, while idf_j refers to the number of documents that use the term j, df_j. It is calculated as

$$idf_j = \log_2(\frac{2 \cdot M}{df_j}).$$

This can be explained as following: the term with high frequency in the text has large weight only if it occurs in a small number of documents. If a high frequency term occurs in many documents, then it does not convey real information because it does not allow for distinguishing between the documents.

The idea that we present in the paper is based on the fact that the terms with medium frequency should have the major weight, and while the terms are more distant from the range of medium frequencies, the less their weight is. There are descriptions of experiments that demonstrate that the usage of variations of weights (Eq. 1) using threshold and transition point (TP) is promising [7] [4]. In this paper, we propose a formula for determining medium frequencies without a threshold. It is shown that this formula allows for obtaining equal or better performance than the TP.

The work has the following structure: first we describe how to determine the range of medium frequencies of a text, then we present the suggested weighting formula, after this, the extraction of the representative sentences from a text is described, and finally, the results of the experiments are discussed.

2 Transition Point Range

One of the important tasks of text representation is the selection of a subset of terms that are good representation of a text and permit operations of categorization, clustering, searching, etc. using the selected subset instead of a whole document. There are various methods for selection of indexing terms or key words, for example, Urbizagástegui [2] used the transition point for showing the usefulness of text indexation.

Transition point is a frequency of a term that divides text vocabulary into terms of low and high frequencies. The terms that are useful for text representation are situated around the TP, because it is supposed that they have high semantic content. The formula for the calculation of TP is as follows:

$$TP = \frac{\sqrt{1 + 8 * I_1} - 1}{2}, \tag{2}$$

where I_1 represents the number of terms with frequency 1. This empiric formula seeks the identification of a frequency that is neither low, nor high. Usually, many terms correspond to low frequencies; say, more that 50% of terms in an average text have frequency 1, etc. This formula excludes them from the consideration explicitly. The calculated frequency (TP) is the lowest of the high frequencies. The alternative calculation of TP is seeking the lowest frequency that is not repeated, i.e., the first frequency that corresponds to exactly one

term. It is justified by the fact that several terms usually correspond to values of low frequencies.

In this paper, we suggest to use two transition points basing on the idea of repetition of frequencies. The first TP is the lowest of the high frequencies that is repeated, TPa, i.e., we start form the highest frequency and go downwards until we find the first repetition. The second TP is the highest of the low frequencies that is not repeated, TPb, i.e., we start from the lowest frequency and go upwards until we find the first term with unique (non-repeated) frequency. Thus, we define the range of medium frequencies, namely, *transition range*, $[TPb, TPa]$.

In the following sections, we describe the application of the transition range to information retrieval and extraction of the representative sentences tasks.

3 Weighting of Terms

As we mentioned before, the documents can be represented by the weighted terms. In [7] a scheme of term weighting which takes into account the TP is presented. The method proposed there is different from Eq. 1, namely

$$d_{ij} = IDPT_{ij} \times DPTC_i, \tag{3}$$

where $IDPT_{ij} = 1/|TP_j - tf_{ji}|$ is the inverse distance of the term i to the TP of the document j, and $DPTC_i = |TP - fr_i|$, is the distance between the term i and TP, calculated for the whole collection.

For our experiment, the definition of $IDPT_{ij}$ is given by:

$$IDPT_{ij} = \begin{cases} 1 & \text{if } tf_{ji} \in [TPb_j, TPa_j], \\ 1/(tf_{ji} - TPa_j) & \text{if } TPa_j < tf_{ji}, \\ 1/(TPb_j - tf_{ji}) & \text{if } TPb_j > tf_{ji}. \end{cases} \tag{4}$$

where $[PTb_j, PTa_j]$ is the transition range of the document j. $DPTC_i$ also is adapted to the two frequencies of transition that are global now:

$$DPTC_i = \begin{cases} 1 & \text{if } fr_i \in [TPb, TPa], \\ fr_i - TPa & \text{if } TPa < fr_i, \\ TPb - fr_i & \text{if } TPb > fr_i. \end{cases} \tag{5}$$

Here $[TPa, TPb]$ constitutes the transition range of the whole collection and fr_i is the frequency of term i in the collection.

4 Representative Sentences in Texts

Let us consider the task of selection of the most "representative" sentences of a text. We base on the work [8], where the terms near TP are considered for assigning scores to sentences and generate an extract composed by three sentences with major scores. The proposed approach is as follows:

1. Preprocessing. Document splitting into sentences is performed, taking into account abbreviations, etc. The words from the stop list are eliminated from the sentences. These are words like prepositions, articles, etc.
2. Vocabulary extraction. All terms are extracted and their frequencies are calculated.
3. Transition range. The transition range is calculated according to the procedure described above. The "virtual paragraph" (VP) is generated, i.e., the paragraph, to which all terms that belong to the transition range are added.
4. Assignment of scores to sentences. Each sentence is assigned a score according to its similarity to the VP.
5. Extraction of representative sentences. Three sentences with major scores according to their similarity to the VP are taken.

The extract quality is verified by its comparison with the complete document. One of the ways of doing this is the usage of the extract instead of the full text in certain tasks like Information Retrieval. If the IR system performs in the same way, then the quality of the extract is good. For our experiments, we used Jaccard's formula to calculate the similarity between the query and each document in the collection, as in [8].

$$sim(D, q) = \frac{\#(D \cap q)}{\#(D \cup q)}.$$

5 Obtained Results

We conducted experiments with term weights assignment based on the transition range and detection of representative sentences. Several subcollections of TREC-5 were used allowing comparison of results with previous works. Further we describe subcollections and then the obtained results.

5.1 Data Description

Collection TREC-5 is a text collection of more than 50,000 documents in Spanish and 50 topics (possible queries). Each topic is assigned a set of documents that correspond to it, i.e., are relevant for this topic. The TREC-5 documents, queries, and relevance criteria were used in our experiments. We defined three subcollections from the documents according to the following algorithm: for a given topic, we add all relevant documents to the subcollection, and then add twice as many non-relevant documents. Table 1 contains the number of documents in subcollections.

The subcollections were preprocessed and words from stop lists were eliminated. The queries were preprocessed as well in the same way. Besides, all letters in queries were changed to lower case. The topics are shown in Table 2.

Table 1. TREC-5 subcollections for 6 topics.

Subcollection	Topics	#	#Relevant
1	$c_1 : c_3$	1117	211 : 164
2	$c_{10} : c_{11}$	933	206 : 105
3	$c_{14} : c_{15}$	817	281 : 6

Table 2. Topics used in evaluation.

c1	mexican oposition FTA (free trade agreement)
c3	pollution mexico city
c10	mexico important country transit war antidrug
c11	water rights rivers frontier region mexico unites states
c14	monopoly oil pemex has great influence mexico
c15	dispute fishing caused capture fishing ships unites staes

5.2 Results

In Fig. 1, the results are presented for each subcollection and for each method. The Column 1 refers to the method. For three first rows, the method based on weighting was used, while for three last rows the extract generation was applied. For each subcollection, we calculated the values of precision P, recall R, and F_1 measure, see, for example, [3].

$$P = \frac{\#\text{relevant documents obtained by the system}}{\#\text{total documents obtained by the system}}, \tag{6}$$

$$R = \frac{\#\text{relevant documents obtained by the system}}{\#\text{total relevant documents}}, \tag{7}$$

$$F_1 = (2 \cdot P \cdot R)/(P + R). \tag{8}$$

The methods that are referred to as TR use transition range, while those referred to as TP are based on transition point as it is explained in sections 3 and 4.

Fig. 1. Transition range.

Method	Subcol. 1			Subcol. 2			Subcol. 3		
	P	R	F_1	P	R	F_1	P	R	F_1
Classic	0.28	0.61	0.38	0.21	0.74	0.33	0.33	0.93	0.48
TR	0.24	0.17	**0.2**	0.17	0.2	**0.18**	0.34	0.78	**0.47**
TP	0.34	0.06	0.1	0.13	0.29	0.18	0.44	0.31	0.33
Full text	0.16	0.47	0.24	0.17	0.68	0.27	0.18	0.69	0.28
TR	0.16	0.22	**0.19**	0.19	0.39	**0.26**	0.18	0.48	**0.26**
TP	0.37	0.07	0.11	0.19	0.33	0.24	0.19	0.19	0.19

6 Conclusions

We presented an approach that allows for detection of the transition range, i.e., the range of terms with medium frequencies in a text. This range has the properties that correspond to the expected behavior of the terms, which are in the transition from terms with low frequency to terms with high frequency. It is supposed that terms in this range are the most representative terms of a text. The advantage of the approach is that it does not require choosing manually any thresholds. Certainly, the results are not as good as in the classic approach that uses $tf_{ij} \cdot ifd_j$ or as in the case of usage of the complete documents. We showed that the transition range gives better results than the transition point; however, this claim must be tested in a larger collection. It is necessary to take into account that transition range has similar behavior and inherits practically all numerous applications of the transition point. So, we can recommend the usage of the transition range in natural language processing applications instead of the transition point because of its extensive advantages.

References

1. Salton, G., Wong, A. & Yang, C.: A Vector Space Model for Automatic Indexing, *Communications of the ACM*, 18(11) pp 613-620, 1975.
2. Urbizagástegui, A.R.: Las Posibilidades de la Ley de Zipf en la Indización Automática, `http://www.geocities.com/ResearchTriangle /2851/RUBEN2.htm`, 1999.
3. van Rijsbergen, C.J.: *Information Retrieval*. London, Butterworths, 1999.
4. Moyotl, E. & Jiménez, H.: An Analysis on Frecuency of Terms for Text Categorization, *Proc. of SEPLN-04, XX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural*, pp 141-146, 2004.
5. Moyotl, E. & Jiménez, H.: Enhancement of DPT Feature Selection Method for Text Categorization, *Proc. of CICLing-2005, Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, pp 706-709, 2005.
6. Baeza-Yates, R.: *Modern Information Retrieval*, Addison Wesley, 1999.
7. Rubí Cabrera, David Pinto, Darnes Vilariño & Héctor Jiménez: Una nueva ponderación para el modelo de espacio vectorial de recuperación de información, *Research on Computing Science* 13, pp 75-81, 2005.
8. Claudia Bueno, David Pinto & Héctor Jiménez: El párrofo virtual en la generación de extractos, *Research on Computing Science* 13, pp 83-90, 2005.