Technische Universität Berlin

Forschungsberichte
der Fakultät IV – Elektrotechnik und Informatik

Smart Systems and their Applications

Bernd Mahr · Sheng Huanye
Editors

Proceedings of the 9th International Workshop of
Shanghai Jiao Tong University and
Technische Universität Berlin
held in TU Berlin,
Berlin, Germany, October 14-15, 2010

Bericht-Nr. 2012 – 06
ISSN 1436-9915

Smart Systems and their Applications

Bernd Mahr · Sheng Huanye
Editors

Proceedings of the 9th International Workshop
of Shanghai Jiao Tong University and
Technische Universität Berlin held in TU Berlin,
Berlin, Germany, October 14-15, 2010

Berichtnummer: 2012 / 06
ISSN: 1436 - 9915

Committee

Workshop Co-Chairs

Bernd Mahr
Liqing Zhang

Program Committee

Bao-Liang Lu
Bernd Mahr
Hans-Ulrich Heiß

Organization Committee

Bernd Mahr
Hans-Ulrich Heiß
Uwe Nestmann
Sebastian Möller
Stephan Völker
Thomas Karbe

Editorial Office and Layout

Thomas Karbe
Wolfgang Brandenburg

Book Editors

Bernd Mahr
Technische Universität Berlin
Fakultät IV, Sekretariat FR 3-2
Franklinstr. 28/29
10587 Berlin, Germany

Sheng Huanye
Shanghai Jiao Tong University
Department of Computer Science
and Engineering
Shanghai 20000240, China

Preface

The International Workshop on “Smart Systems and their Applications” is the ninth in a successful series of workshops that was established by Shanghai Jiao Tong University and Technische Universität Berlin. The goal of these workshops is to bring together researchers from both universities in order to present research results to an international community.

The series of workshops started in 1990 with the International Workshop on “Artificial Intelligence” and was continued with the workshop on “Advanced Software Technology” in 1994. Both workshops have been hosted by Shanghai Jiao Tong University. In 1998 the third workshop was organized in Berlin. This workshop was essentially based on results from the Graduiertenkolleg on “Communication Based Systems”, funded by the German Research Society (DFG) from 1991 to 2000. The fourth workshop on “Robotics and its Applications” was again held in Shanghai in 2000. Two years later, in 2002, the fifth workshop on “The Internet Challenge: Technology and Applications” was hosted by TU Berlin, and in 2005 the sixth workshop on “Human Interaction with Machines” was held in Shanghai. The seventh workshop took again place in Berlin. It was devoted to the Topic “Embedded Systems – Modeling, Technology and Application”. At this workshop, for the first time, three students from TU Berlin received their SJTU master degrees and, in addition, their TUB diplomas after studying for two years at the Jiao Tong University under the framework of the dual degree program between SJTU and TU Berlin. Since, more than sixty students from both universities have been awarded the two degrees under this program. In 2008 the eighth workshop on “Autonomous Systems – Self Organization, Management, and Control” was organized at Jiao Tong University. Fourteen guests from TU Berlin participated in this event.

The subject of the ninth workshop on “Smart Systems and their Applications” reflected the recent successes in complex intelligent solutions. The two universities have actively contributed to these, in their fundamental studies as well as in the development of real life applications. This became particularly evident in the invited talk by Ralf-Guido Herrtwich on “Cars and LTE: Beyond the Obvious”. Professor Herrtwich showed in his impressive presentation the path from research to market in smart systems applications in cars. The workshop was chaired by Prof. Zhang, Prof. Heiß and Prof. Mahr and was organized with the help of Prof. Möller from the Telecom Labs at TU Berlin. It showed the high level of quality and international relevance of research and development acquired by the two institutions. And it proved the success of twenty years of cooperation.

Berlin, 25. September 2011

Bernd Mahr and Sheng Huanye

Contents

Additional Papers

The following talks where presented at the workshop, but have been published elsewhere.

Modern Statistical Techniques for Subject-Indepedent BCI decoding

Siamac Fazli

Published in:

Fazli, S., Mehnert, J., Curio, G., Villringer, A., Müller, K.R., Steinbrink, J., Blankertz, B.: Enhanced performance by a hybrid NIRS-EEG Brain Computer Interface. NeuroImage (2011), in press

and

Fazli, S., Popescu, F., Danóczy, M., Blankertz, B., Müller, K.R., Grozea, C.: Subject independent mental state classification in single trials. Neural Networks 22, 1305–1315 (2009)

Cars and LTE: Beyond the Obvious

Ralf Guido Herrtwich

Learning on Structured Representations

Johannes Jain, Klaus Obermayer

Parallel Dataflow Programming Beyond Map/Reduce

Volker Markl, Odej Kao

Published in:

Battré, D., Ewen, S., Hueske, F., Kao, O., Markl, V., Warneke, D.: Nephele/pacts: a programming model and execution framework for web-scale analytical processing. In: Proceedings of the 1st ACM symposium on Cloud computing. pp. 119–130. SoCC '10, ACM, New York, NY, USA (2010), http://doi.acm.org/10.1145/1807128.1807148

Multimodal Interaction

Sebastian Möller, Christine Kühnel, Benjamin Weiss, Ina Wechsung

Published in:

Kühnel, C., Weiss, B., Möller, S.: Parameters describing multimodal interaction ? definitions and three usage scenarios. In: Proceedings of the 11th Annual Conference of the ISCA (Interspeech 2010). pp. 2014–2017. International Speech Communication Association (ISCA) (Sep 2010)

and

Wechsung, I., Naumann, A., Möller, S.: The Influence of the Usage Mode on Subjectively Perceived Quality. In: Spoken Dialogue Systems for Ambient Environments. Proceedings of Second International Workshop, IWSDS 2010, Gotemba, Shizuoka, Japan, October 1-2, 2010. Lecture Notes in Computer Science, vol. 6392, pp. 188–193. Springer (October 2010) and

Wechsung, I., Schleicher, R., Möller, S.: How context determines perceived quality and modality choice. secondary task paradigm applied to the evaluation of multimodal interfaces. In: Proc. IWSDS2011 Workshop on Paralinguistic Information and its Integration in Spoken Dialogue Systems. pp. 239–249 (Sept 1-3 2011)

and

Weiss, B., Kühnel, C., Wechsung, I., Fagel, S., Möller, S.: Quality of talking heads in different interaction and media contexts. Speech Communication (52), 481–492 (2010), http://www.sciencedirect.com/science/article/B6V1C-4YDYSBJ-4/2/fbaf3640c8455fedeb74070

A Quasi-Intelligent System for Constructing Chinese Opinion-Element Collocation Database using Search Engine and HowNet

Mosha Chen[1] , Tianfang Yao[1]

[1] Department of Computer Science and Engineering
Shanghai Jiaotong University, China
mosha@sjtu.edu.cn; tf-yao@cs.sjtu.edu.cn

Abstract. In this paper, we show a demo system and present a novel approach for constructing a collocation database for the Chinese language. A collocation is a pair of two words that are most likely to co-exist in both verbal and written language, while an opinion-element collocation is a collocation whose composition words contain opinion/sentiment element. An opinion-element collocation database will be useful for the Opinion Mining task in many aspects. We use search engine as our fundamental tool mainly because it could help us to seek both domain-specific and domain-independent collocation pair, and we use HowNet as a resource because it can offer rich semantic information which will help us to classify the collocation into domain-specific or domain-independent type. The tool and resource are combined to build a smart system that can automatically crawl data from the Internet and analyze the extracted collocation. Finally, in order to ensure the quality of the extracted collocation we evaluate it manually. The experimental results on the COAE2008's public corpus and the opinion-element collocation database produced by our system have proved the success of this approach for the four domains.

Keywords: collocation, opinion mining, search engine, HowNet

1 Introduction

Opinion Mining is a hot research task in recent years [1, 2, 3, 4, 5, 6], which deals with the traditional NLP research area but contains many advanced sentiment analysis technologies. According to Kim and Hovy's definition [1], an opinion contains four elements: claim, holder, topic and sentiment. In a claim, a holder (may be not existing) has sentimental comments on some topic (or more than one topic). Since the final aim of opinion mining is to get the sentimental polarity of a specified topic or text, and the polarity of a topic is determined by the sentiment that modifies it. Therefore, topic and sentiment play an essential role in opinion mining task and most of the recent work [1, 3, 4, 7, 8, 9, 10] focuses on topic and sentiment, including both English and other languages.

A sentence(separated by a period) may contain multiple topics and multiple sentiments, and sometimes a sub-sentence(separated by a comma) may also contain multiple topics and multiple sentiments, depending on the special grammar of the language itself, so it is necessary to point out which sentiment modifies which topic, which is a key point in opinion mining task. Among all the work involving the analysis of topic polarity, there is a methodology that analyzes topic and sentiment from the perspective of collocation [10, 11]. A collocation is a pair of two words that co-exist in both verbal and written language, for example, the word "rich" comes to you, the first thing that comes into your mind is somebody, like "Bill Gates". An opinion-element collocation is one that contains opinion element in its composition words, and in this research we limit its composition to be a topic word and a sentiment word. Like, "smart" and "system" form an opinion-element collocation. An opinion-element collocation is more suitable for the Opinion Mining task because sometimes it could help determine and validate the precious matching of a given topic and its corresponding sentiment especially when there are multiple topics and sentiments in a sentence.

In this paper, a novel approach is applied to construct such an opinion-element collocation database. It is constructed in the form of a pipeline: firstly we use an online search engine to crawl and collect the intended data from the Internet; secondly, after data cleaning and preprocessing, the crawled data are transferred to the next stage to be analyzed by a dependency parsing tool Deparser [12], which can help extract the potential topic-sentiment collocations; Finally, HowNet [13], functioning like WordNet [14], is used to mine the semantic information behind the collocation pairs as well as to classify the pairs into domain-specific and domain-independent ones. Because the whole system also needs human involvement and intervention, for example, human judgment is needed to check whether the potential collocation extracted by Deparser is correct or not, therefore we call our system quasi-intelligence system, that is, self-automatic. Section 4 gives the detailed overflow of the whole system. The system is established to construct an opinion-element collocation resource for more advanced research and is served as a basic component. This paper focuses on how we construct the system and some design details about the system, more application can be found in a recent research [11]. The experiments is conducted on the COAE2008's public corpus [15] to compare results before and after using the knowledge of opinion-element collocation and the experimental result proves that the collocation database can help a lot in our approach.

The following of this paper is organized as follows: Section 2 talks about the related work. Section 3 introduces the tools and resource we utilize and Section 4 gives the overall architecture of our system and gets into the details of each component. Section 5 presents the experiments and the conclusion is given in Section 6.

2 Related Work

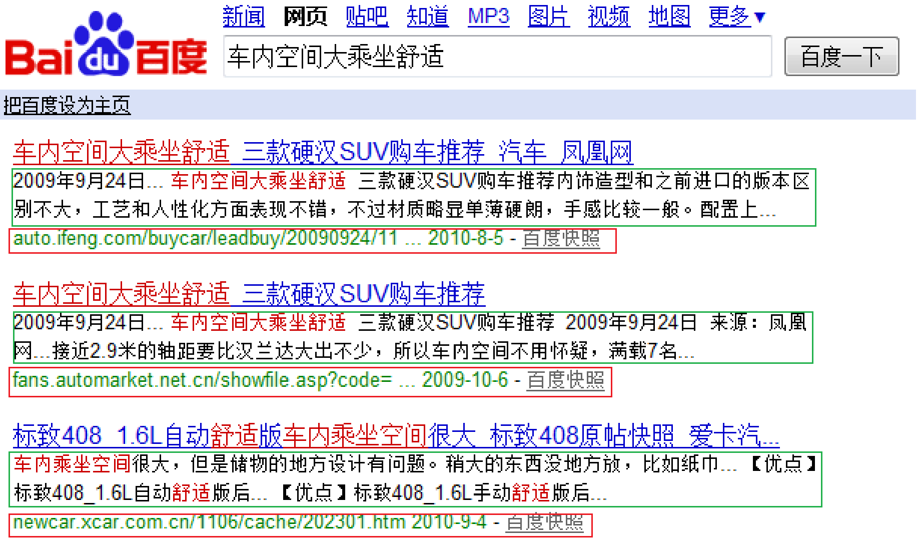
There are many topics discussing about corpus construction in recent years; there is a trend that researchers begin to facilitate the online resource to conduct their experiments. For example, Wikipedia [16] is a good resource for constructing ontology; Google [17] is famous for its large-scale data and many researchers have used the massive data to conduct experiments and test their ideas. In Chinese, we also have many function-like online resources, like Sogou Lab [18] and Baidu [19], of which the former offers many useful corpus extracted from the Internet and the latter is a good search engine especially for Chinese language processing. We list two researches [20, 21] which have taken search engine as a tool for related collocation researches, and we believe the Internet will play a more and more important role in both research and life. It's not the first time that we use search engine for research or the last time.

3 Tools and Resource

In this section, we briefly introduce the tools and resource utilized in our system.

3.1 Search Engine

Here we use search engine mainly to get the search snippet for each search item in the search result lists. Fig.1 gives a screenshots of a search result.



车内空间大乘坐舒适_三款硬汉SUV购车推荐_汽车_凤凰网
2009年9月24日... 车内空间大乘坐舒适 三款硬汉SUV购车推荐内饰造型和之前进口的版本区别不大，工艺和人性化方面表现不错，不过材质略显单薄硬朗，手感比较一般。配置上...
auto.ifeng.com/buycar/leadbuy/20090924/11 ... 2010-8-5 - 百度快照

车内空间大乘坐舒适_三款硬汉SUV购车推荐
2009年9月24日... 车内空间大乘坐舒适 三款硬汉SUV购车推荐 2009年9月24日 来源：凤凰网...接近2.9米的轴距要比汉兰达大出不少，所以车内空间不用怀疑，满载7名...
fans.automarket.net.cn/showfile.asp?code= ... 2009-10-6 - 百度快照

标致408_1.6L自动舒适版车内乘坐空间很大_标致408原帖快照_爱卡汽...
车内乘坐空间很大，但是储物的地方设计有问题。稍大的东西没地方放，比如纸巾... 【优点】标致408_1.6L自动舒适版后... 【优点】标致408_1.6L手动舒适版后...
newcar.xcar.com.cn/1106/cache/202301.htm 2010-9-4 - 百度快照

Fig. 1. Search Result for "车内空间大乘坐舒适(the inner space of this car is large and it sits comfortable)" using Baidu. Content in green box represents the snippet and words in red means hitting the keyword. Content in red box represents the URL where the search result comes from, the URL domain will be counted and the frequently existing ones will be kept for future use, i.e. they may be domain-specific sites.

The terminology "snippet" can be interpreted as the abstract for the page content. The snippet tries to summarize and cluster similar and related sentences into it to meet the users’ purpose. So it is reasonable that the snippet contains useful and meaningful information for the given key words. In our operation, we input a sentence from the testing corpus and want to get similar context, it is effective to use the snippet for such a task.

3.2 Deparser

Deparser is a NLP tool that can be used for many Chinese NLP tasks, for example, word segmentation, POS tagging, dependency parsing and so on. In this section, the dependency parsing is introduced because we depend on this module to extract the potential opinion-element collocations. The dependency parsing algorithm takes some strategies to extract the most probability modifying-modified dependency relations in a sentence, so it is also an intuitive idea to get the intended opinion-mining collocation from the dependency pair plus the POS and dependency type information [7], if such opinion-element collocation does exist in the analyzed sentences and satisfy certain conditions [7]. Detailed algorithm is given in Section 4. Fig.2 gives a simple example of dependency result from which you could get a direct impression of how Deparser works.

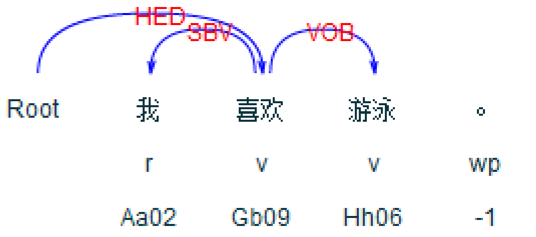


Fig. 2. Dependency Analysis Result for “我喜欢游泳 (I like swimming)” using Deparser. In this example, “我（I）” and “喜欢(like)” form a SBV(subject-verb) dependency; “喜欢(like)” and “游泳(swimming)” form a VOB(verb-object) dependency. “喜欢(like)” and “游泳(swimming)” form an opinion-element collocation because “喜欢(like)” is a topic element and “游泳(swimming)” is a sentiment element.

3.3 HowNet

HowNet is an online common-sense knowledge base unveiling inter-concept relations and inter-attribute relations of concepts as connoting in lexicons of the Chinese and their English equivalents [13]. It offers rich information for the lexicon, including some semantic information, like the different meaning of a word and the occasions when it is used in. Take the word “品尝(taste)” for example, it has the following property: DEF=attribute|属性,taste|味道,&edible|食物. The DEF property gives the concept and attributes of a word (lexicon). In scenario “taste some foods”, “taste” plays as a verb; in scenario “the taste is not bad”, however, “taste” plays as a noun. Another example is that “doctor” and “patient” are belong to a more general concept “human being”. We will take advantage of the hierarchy information supplied by HowNet to classify the opinion-element collocation into domain-specific and domain-independent types.

4 System Introduction

This section gives the overall architecture of the system and goes into details for each component.

4.1 Architecture of the System

Fig.3. presents the architecture of the whole system. The system can be divided into four parts: Crawler, Parser, Human Editor and Classify Component. Crawler is the component using search engine to get the snippet for each query. Parser is the component used to extract the potential opinion-element collocations. Human Editor is for manual correction and modification, one more notice is that the Human Editor module is added in this system for quality issue. In our previous studies, it doesn’t exist because we approximately regard the potential opinion-element collocations extracted by Deparser is correct. The classifier adopts HowNet to classify the collocations into domain-specific and domain-independent types for further use. The interchange files are all formatted in self-defined XML schema.

4.2 Crawler

The crawler gets user input and sends the http request to the specified search engine using its public API, and then it processes the received data to get the snippet for each query. Also some data cleaning and preprocessing work should be completed in this component. Fig.4. shows a snapshot for this component. The input sentences are selected directly from the corpus used for other advanced researches. We benefit from the snippet because it can return similar and related information except for the original query. As to data cleaning and preprocessing, two kinds of information are considered in this paper: replication and advertisement. Replica is caused mainly because the same article is reshipped by many sites and it is easy to kick off the replica by computing the similarity of the snippet. For ads, a list of common advertisement words, like “discount”, “for sale” is collected and we can discard the snippet if it hits too many obvious advertisements. How many snippets should we collect? By experience, we choose the first 5 search result pages as our sources for snippet because the quality for the search is not satisfied due to our observation on average. Of course this value can be modified in the configuration file.

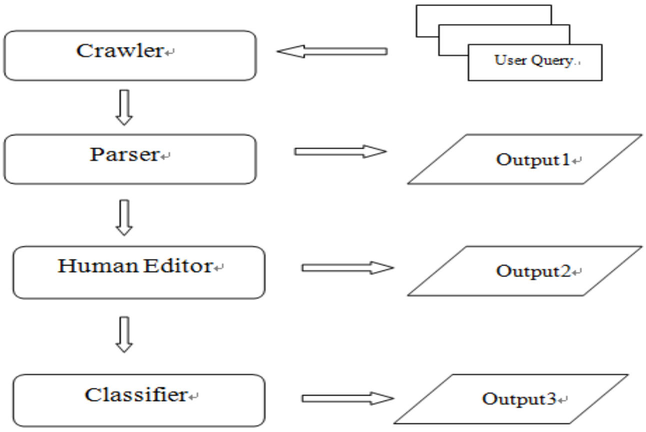
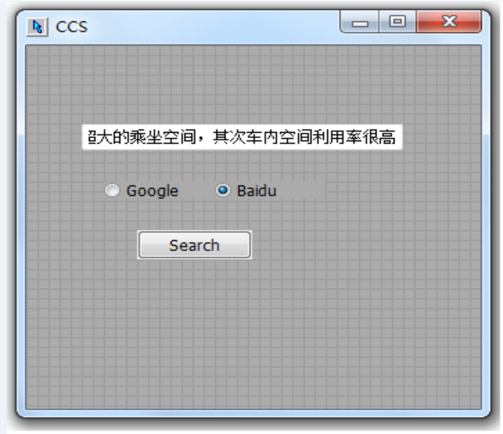


Fig. 3. System Architecture. The system starts from user query input query, then it makes use of search engine to crawl the search snippet, after parsing, human editor and classifying processing, we get the final collocation database. After Parser and Human Editor, there is also output of collocations, compared to the final collocation, they are of different use.



召大的乘坐空间，其次车内空间利用率很高

○ Google ● Baidu

Fig. 4. Snapshot for the Crawler Component. We choose Baidu as the default search engine because it is more professional in Chinese.

4.3 Parser

The Parser component analyzes and extracts the potential collocations from the output of the Crawler component. In the previous work [7], we find three types of dependency relations are important for Opinion Mining task, that is SBV (subject-verb), VOB (verb-object) and ATT (attribute). For example, in the sentence “The boy is cute", the words "boy" and "cute" form an SBV dependency relation; in the sentence "I like swimming", the words "like" and "swimming" form a VOB dependency relation; in the sentence "beautiful pictures and delicious food", the words "beautiful" and "pictures" form an ATT relation. These three dependency types cover almost 90% of the opinion-element relations in the corpus we conduct experiments on, so it is reliable that we only consider these three types to extract the opinion-element relations, which can be regarded as the potential opinion-element collocations. Additionally, we examine whether a collocation contains sentimental word in it to determine whether it is a potential opinion-element collocation. For efficiency issue, we can start multiple threads for the Parser component since it is the most time-consuming module in the whole system. The value could be modified in the configuration file. Below is a simplified algorithm from research [7] that we apply to extract the potential collocations.

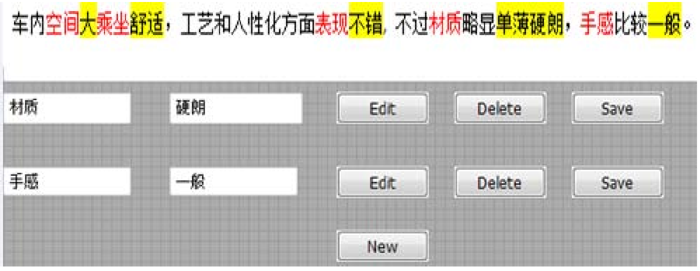
```
Input: Sentence S, Sentiment Dictionary SentDict
Output: Collocation Set ColSet
program ExtractPotentialOpinionElementCollocation:
ColSet = {}
DepRelationSet = Parse(S)
Foreach DepRelation in DepRelationSet:
  If DepRelation in {SBV,ATT,VOB}:
    (word1, word2) = DepRelation
    If word1 or  word2 in SentDict:
      ColSet.Add(DepRelation)
```

4.4 Human Editor

This component is designed for human editing the result conducted by the Parser component because the opinion-element collocation extracted by Deparser may have errors especially when there are multiple topics and sentiments in a single sentence. So it is necessary to conquer this issue with a human editing process. Fig. 5 gives a screenshot for this component.

4.5 Classifier

This component classifies the extracted collocations into domain-specific and domain-independent ones for further use. Intuitively speaking, the word "good" can almost modify any object, but some words are domain-specific, like "fuel-consuming" is most probably specified to the engine. HowNet offers rich information for each lexicon it includes. For each collocation, we focus on both the sentiment word and the topic word, by examining the values given in the DEF property and the hierarchy information it offers, we can classify them to the specified domains. As you could image, HowNet can't promise every word having its definition, so the ones that don't exist in HowNet are labeled as unknown. Due to our subsequence work, now we have four domains: car, digital camera, PC and MP3, plus the domain-independent and unknown class, there are 6 classes in all. Randomly picking items in each domain-specific class, we find the classification effect is to our satisfaction.



车内空间大乘坐舒适，工艺和人性化方面表现不错, 不过材质略显单薄硬朗，手感比较一般。

材质 硬朗

手感 一般

Fig. 5. Snapshot for the Human Editor Component. The colored line is the extracted result from the Parser component. Words in red present topic and words in yellow present sentiment. The following table is the extracted potential opinion-element collocations, you can “edit”, “delete” or “save” a potential collocation, even you can “new” a collocation if it is missed.

5 Experiments

We conduct experiments from two perspectives, one is to test the system itself to prove our approach is feasible and adaptive; the other is to use the collocations extracted from the system for further research to prove the collocation database is useful and valuable.

5.1 Corpus

The testing corpus is from COAE2008’s public testing corpus. COAE [15] stands for Chinese Opinion Analysis Evaluation, so it is proper as a base to get more corpuses from the Internet. The corpus has four domains, that is, car, mobile, PC and MP3. Each domain contains about 100 articles and each article contains 1 to 7 sentences. Each article is considered to be of sentimental polarity, but the sentences it contains may be declarative sentences. The following is an example from the public corpus: “哈弗 M2 的座椅为双色织布面料，与内饰整体色调比较协调，侧向支撑力不错。驾驶员座椅为手动四向调节，可满足各种身材、体形的驾驶者，提供最佳驾驶姿态。(Harvard M2 has seats with double-dyed fabric, which are harmony with the overall tone of the internal decoration; the lateral support is satisfying. The driver’s seat can manually adjust to four directions, which is suitable for drives of different shapes and can offer the best driving posture)”. Words in yellow means the sentiment word, you may notice that some sentences don’t contain sentiment word at all, so most probably it is just a declaration. We will select the sentences that contain sentiment word to be the input query for our system.

5.2 Experiments on System

To our knowledge, there was no similar work on the construction of collocation database before, so we try to evaluate and test from the following aspects:

- Output/Input Rate:
  Here the “Output/Input Rate” is computed as Eq.1:

  $$\text{O/I-Rate} = \frac{\#\,(number\ of\ extracted\ collocations)}{\#(number\ of\ input\ query)} \quad (1)$$

  Table 1 shows the O/I Rate according to different input query. The input query sentences are randomly selected from the COAE2008’s public corpus and the extracted collocations are directly extracted after the Parsing component, in which no human intervention is involved. The rate decreases as the number of input query increases. It is because that the input sentences may come from the same article in the corpus, so the search result returned may be similar, so there is duplication. But as you can image, this simple but novel approach could indeed get the intended collocations, 200 input query sentences could get as many as nearly 7000 collocation pairs, which is to our delights. Little effort, but great pay!

Table 1. O/I Rate Experiments Results

#(input query)	#(extracted collocation)	O/I Rate
100	3567	35.67
200	6720	33.60
300	8340	27.80
400	10024	25.60
500	10943	21.87
Ave.		28.91

- Human involvement Effect
  This criterion is used to evaluate how the human involvement affects, or in other words, we want to know how the Parser component works. Due to this is a time-consuming work, we just select 20 sentences to compare the results before and after human modification. We make statistic on the #(edit number), #(delete number) and #(new number). Table 2 gives the result:

Table 2. Human Validation Results based on 20 Queries

#(query)	#(collocation)	#(Edit)	#(New)	#(Delete)
20	711	42	17	27
Percentage		5.9%	2.3%	3.7%

As you could notice, the overall human effort affects for 10% about the collocations extracted directly from the Parser component, so we think the results extracted by our system is reliable and the quality is ensured.

- Classification Effect:
  This result depends on the coverage fraction of the HowNet. We extract collocations from 200 sentences in Table 5.2 and classify the collocations into 6 classes (Section 4.5 gives the details). Altogether 6720 collocations and there are nearly 4000 collocations belong to the unlabeled class, the four domains(car, mobile phone, PC and MP3) counts about 1400. Randomly picking collocations from the 4 domains, we feel the classfication result is to our satisfaction. This is just a try, we can complete and mine further on this component, more semantic information should be taken into consideration.

5.3 Experiments on Collocations

We just list the result from one of our previous studies [11] to show the collocations indeed could help in further research. Table 3 gives the result:

Table 3. Opinion-element Relation Extraction Results

	P	R	F
Baseline1(Closest-pair)	51.60%	73.85%	79.60%
Baseline2(Parsing)	72.53%	85.99%	78.63%
Our method(Collocation)	73.92%	93.37%	82.47%

The experiment is conducted to extract the opinion-element relations that exist in a given opinioned sentence. The testing corpus is also the COAE2008's corpus. An opinion-element relation is a relation between a topic word a the sentiment word, the sentiment word modifies the sentiment word. In the Parser component, the extracted dependency relations are such relations if they are correct. From a more general level, an opinion-element relation is an opinion-element collocation, the former is specified to a sentence and the latter is oriented to the statistic of language usage. The collocation plays an important role in this experiment and you could see that the recall of our method obviously improves to the existing common method. The collocation database in this experiment could help seek the opinion-element relations whose composition words may apart from each other, perhaps exist in different sub-sentences, which is beyond the parser's ability to extract. It has proved the success application of the collocation database.

6 Conclusion

The motivation for this work depends on our studies for Opinion Mining, in work [11], we extract opinion-element relations for Chinese; we find it necessary to build such a collocation database to help get a better result, so we did. In work [11], we have proved our method and the collocation database plays an import role in that work. Further, we think it can be extended to other uses, just list some: Chinese spelling check, for some domain-specific sentences, some words are rare and users who use Pinyin IME may get the wrong words printed on screen(the mis-spelled words have the same pronunciation with the correct one, but with different characters), for example, “蓝牙(Bluetooth)” is often mis-spelled as “篮牙”. If we get the context around the mis-spelled word and the content can also be found in the domain-specified collocations, then we believe, we can correct such kinds of mis-spelling via some strategies. Hope we can benefit from this system in future research and some extensions can be made on this system to make it more powerful!

Acknowledgement

This research work is financially supported by the National Science Foundation of China (No. 60773087) and the UDS-SJTU Joint Research Lab for Language Technology. Besides, Information Retrieval Lab, Harbin Institute of Technology provides the Chinese syntactic analyzer LTP Deparser, and KEENAGE offers the HowNet platform. We sincererly thank them for their help.

References

1. S.-M Kim, E. Hovy. Determining the Sentiment of Opinions[A]. In Proceedings of COLING04. The Conference on Computational Linguistics(COLING2004)[C].
2. B Liu, M Hu, J Cheng. Opinion Observer: Analyzing and Comparing Opinions on the Web. Proceedings of the 14th International Conference on World Wide Web.
3. S.-M Kim and E. Hovy. Identifying and Analyzing judgment opinions. Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference(HLT-NAACL), 2006
4. AM Popescu, O Etzioni. Extracting Product Features and Opinions from Reviews. Proceedings of EMNLP 2005[C]. 2005
5. S. Matsumoto, H.Takamura, M.Okumura. Sentiment Classification using Word sub-Sequences and Dependency Sub-trees. Proceedings of PAKDD'05, the 9th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, 2005.
6. B. Pang, L. Lee. Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2008.
7. D.Lou, T. Yao. Semantic Polarity Analysis and Opinion Mining on Chinese Review Sentences. Computer Application. Volume 26, 2006.
8. J.Zhang, Q.Zhang, L.Wu and X.Huang. Subjective Relation Extraction in Chinese Opinion Mining. Volume 22, 2008
9. Q.Chen, Q.Liu and T.Yao. Topic and Sentiment Relation Extraction on Chinese Opinioned Texts. in Proceedings of the National Conference on Information Retrieval(CCIR), pp.505-512, 2009.
10. N. Kobayashi, K. Inui, Y. Matsumoto: Extracting Aspect-Evaluation and Aspect-of Relations in Opinion Mining, EMNLP-CoNLL 2007.
11. M.Chen and T.Yao. Combining Dependency Parsing with Shallow Semantic Analysis for Chinese Opinion-element Relation Extraction. In Proceedings of International Universal Communication Symposium, 2010, (to be published)
12. http://ir.hit.edu.cn/demo/ltp
13. http://www.keenage.com/
14. http://wordnet.princeton.edu/
15. http://ir-china.org.cn/coae2008.html

16. http://en.wikipedia.org/
17. http://www.google.com/
18. http://www.sogou.com/labs/
19. http://www.baidu.com/
20. B.Zengin. Benefit of Google Search Engine in Learning and Teaching Collocations. Eurasian Journal of Educational Research, 34, pp. 151-166, Winter 2009
21. O.Etzioni, M.Banko, S.Soderland and Daniel S.Weld. Open Information Extraction from the Web. Communication of the ACM, 51(12):68-74, 2008

A One-layer Recurrent Neural Network with a Unipolar Hard-limiting Activation Function for Linear Programming with an Application to Linear Assignment

Qingshan Liu[1] and Jun Wang[2]

[1] School of Automation, Southeast University, Nanjing 210096, China
qsliu@seu.edu.cn
[2] Department of Mechanical and Automation Engineering
The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong
jwang@mae.cuhk.edu.hk

Abstract. This paper presents a one-layer recurrent neural network for solving the linear programming problems. The proposed neural network is guaranteed to be globally convergent in finite time to the optimal solutions under a mild condition on a derived lower bound of a single gain parameter. The number of neurons in the neural network is the same as the number of decision variables of the optimization problem. Compared with the existing neural networks for linear programming, the proposed neural network has salient features such as finite-time convergence and a low model complexity. Specifically, the proposed neural network is tailored to solve the linear assignment problem with simulation results to demonstrate the effectiveness and characteristics of the proposed neural network.

Key words: Recurrent neural network, linear programming, global convergence in finite time, linear assignment

1 Introduction

Consider a general linear programming problem as follows:

$$\begin{array}{ll} \text{minimize} & c^T x, \\ \text{subject to} & Ax \le b. \end{array} \tag{1}$$

where $x = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n$ is the decision vector, $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$.

Linear programming has widespread application scope in science and engineering, such as associative memory, linear assignment and shortest path routing.

The work described in the paper was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China, under Grants CUHK417608E and CUHK417209E.

Over the past two decades, recurrent neural networks as parallel computational models for optimization have received substantial attention [1][2]. Specially, recurrent neural networks for linear programming have been well developed. In 1986, Tank and Hopfield [1] proposed a recurrent neural network for solving the linear programming problems for the first time. In 1988, the dynamical canonical nonlinear programming circuit (NPC) was introduced by Kennedy and Chua [2] for optimization by utilizing a finite penalty parameter, which can generate the approximate optimal solutions. Wang [3][4][5] proposed some primal-dual, primal, and dual neural networks for solving the shortest path and linear assignment problems. Xia [6] proposed a primal-dual neural network for solving the linear programming problems. Forti et al. [7] investigated the generalized NPC (G-NPC) for non-smooth optimization, which can be considered as a natural extension of NPC. In order to reduce the model complexity, we proposed a one-layer recurrent neural network for solving the linear programming problems, which had lower model complexity [8].

Recently, the finite penalty parameter method was introduced for solving optimization problems with bounded feasibility region [7]. However, in many applications, such as linear assignment [4][9] and shortest path problems [5][10][11], the feasibility region of the optimization problems are mostly unbounded. Then the neural network in [7] is not capable of solving these problems. This paper is concerned with a one-layer recurrent neural network for linear programming, in which the feasibility region can be unbounded. As a result, the proposed neural network is capable of solving more general linear programming problems. In addition, the global convergence of the recurrent neural network is guaranteed to be in finite time provided that its gain parameter in the nonlinear term exceeds a given lower bound. Moreover, the proposed neural network is utilized to solve the linear assignment problems.

2 Model Description

This section describes the recurrent neural network model for solving optimization problem (1). According to the Karush-Kuhn-Tucker (KKT) conditions [12], x^* is an optimal solution of problem (1), if and only if there exists $y^* \in \mathbb{R}^m$ such that

$$c + A^T y = 0, \tag{2}$$

$$\begin{cases} y_j \geq 0, & \text{if } a_j x - b_j = 0, \\ y_j = 0, & \text{if } a_j x - b_j < 0, \end{cases} \tag{3}$$

where y_j and b_j are the jth components of y and b respectively, and a_j denotes the jth row of A $(j = 1, 2, \ldots, m)$.

Based on (2) and (3), the dynamic equation of the proposed recurrent neural network model is described as follows:

$$\epsilon \frac{dx}{dt} = -\sigma A^T g(Ax - b) - c, \tag{4}$$

where ϵ is a positive scaling constant, σ is a nonnegative gain parameter, $g(v) = (g_1(v), g_2(v), \ldots, g_m(v))^T$ is the unipolar hard-limiting activation function and its component is defined as

$$g_j(v) = \begin{cases} 1, & \text{if } v > 0, \\ [0,1], & \text{if } v = 0, \quad (j = 1, 2, \ldots, m) \\ 0, & \text{if } v < 0. \end{cases} \tag{5}$$

3 Convergence and Optimality Analysis

In this section, the finite-time global convergence of the proposed neural network is presented, and the optimal solution of problem (1) is guaranteed using the proposed neural network with sufficiently large gain parameter σ over a derived lower bound. In this paper, we denote $\mathit{\Gamma} = \{\gamma : \gamma = (\gamma_1, \gamma_2, \ldots \gamma_m)^T \in \mathbb{R}^m, 0 \leq \gamma \leq 1$ and at least one $\gamma_i = 1\}$ and $A^T\mathit{\Gamma} = \{A^T\gamma : \gamma \in \mathit{\Gamma}\}$.

Let $\psi(x) = c^Tx + \sigma e^T(Ax - b)^+$, where $e = (1, 1, \ldots, 1)^T \in \mathbb{R}^m$ and $v^+ = \max\{0, v\}$. Then its generalized gradient is $\partial\psi(x) = c + \sigma A^T K[g(Ax - b)]$, where $K(\cdot)$ denotes the closure of the convex hull. Then the neural network in (4) is a gradient system of energy function $\psi(x)$. Since $\psi(x)$ is convex, the minimum point of $\psi(x)$ corresponds to the equilibrium point of neural network (4). Thus, if neural network (4) has a stable equilibrium point, then the minimum point of energy function $\psi(x)$ is guaranteed. Next, we give the finite-time global convergence of the proposed neural network as follows.

Theorem 1. *If $\psi(x)$ has a finite minimum, then the state vector of neural network* (4) *is globally convergent to an equilibrium point in finite time with any $\sigma \geq 0$.*

Proof: From the assumption, the equilibrium point set of neural network (4) is not empty. Denote $\bar{x}$ as an equilibrium point of neural network (4), then it is a minimum point of $\psi(x)$. It follows that $0 \in \partial\psi(\bar{x})$.

Consider the following Lyapunov function:

$$V(x) = \epsilon(\psi(x) - \psi(\bar{x})) + \frac{\epsilon}{2}(x - \bar{x})^T(x - \bar{x}), \tag{6}$$

We have

$$\partial V(x) = \epsilon(\partial\psi(x) + x - \bar{x}).$$

Similar to the proof of Theorem 3 in [13], we have

$$\dot{V}(x(t)) \leq - \inf_{\eta \in \partial\psi(x)} \|\eta\|_2^2, \tag{7}$$

and the state vector of neural network (4) is globally convergent to an equilibrium point.

Next, we prove that the convergence is in finite time. Assume $x(t)$ is not an equilibrium point, so that $0 \notin \partial\psi(x)$. Since $\partial\psi(x) = c + \sigma A^T K[g(Ax - b)]$

is nonempty and compact, one gets that $\inf_{\eta\in\partial\psi(x)} \|\eta\|_2^2$ is a positive constant, denoted as β. Then, from (7), we have

$$\dot{V}(x(t)) \leq -\beta.$$

Integrating both sides of the above inequality from 0 to t, it is easily to verify that $V(x(t)) = 0$ for $t \geq V(x(0))/\beta$. From (6), since $V(x(t)) \geq \epsilon\|x - \bar{x}\|_2^2/2$, we get that $x = \bar{x}$ for $t \geq V(x(0))/\beta$. That is, $x(t)$ is globally convergent to an equilibrium point in finite time. □

From the above analysis, the proposed neural network is guaranteed to reach an equilibrium point in finite time. To obtain the optimal solution of problem (1), the relationship between the optimal solution of problem (1) and the equilibrium point of neural network (4) is presented as follows.

Theorem 2. *Any optimal solution of problem* (1) *is an equilibrium point of neural network* (4) *and vice verse, if*

$$\sigma > \frac{\|c\|_2}{\min_{A^T\gamma\in\Delta} \|A^T\gamma\|_2}, \tag{8}$$

where Δ is the largest compact set in $A^T\Gamma\backslash\{0\}$.

Proof: The proof is omitted due to limitation of space. □

According to Theorems 1 and 2, the state vector of neural network (4) is globally convergent to an optimal solution of problem (1) in finite time if (8) holds.

4 An Application for Linear Assignment

In this section, the proposed neural network is utilized for solving the linear assignment problems. In the literature, several recurrent neural networks have been developed for linear assignment (e.g., see [4][9]). The general assignment problem is to find an optimal solution to the following linear integer programming problem:

$$\text{minimize} \quad \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}, \tag{9}$$

$$\text{subject to} \quad \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, 2, \ldots, n, \tag{10}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, 2, \ldots, n, \tag{11}$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, 2, \ldots, n. \tag{12}$$

According to [14], if the optimal solution of the primal assignment problem (9)-(12) is unique, then it is equivalent to a linear programming problem by replacing the zero-one integrality constraints (12) with nonnegative constraints as follows:

$$x_{ij} \geq 0, \quad i, j = 1, 2, \ldots, n. \tag{13}$$

Based on the primal assignment problem, the dual assignment problem can be formulated as follows:

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^{n}(u_i + v_i), \\ \text{subject to} & u_i + v_j \leq c_{ij}, \quad i, j = 1, 2, \ldots, n, \end{array} \tag{14}$$

where u_i and v_i denote the dual decision variables.

Let $x = (u_1, \ldots, u_n, v_1, \ldots, v_n)^T$, $c = (-1, -1, \ldots, -1)^T \in \mathbb{R}^{2n}$, and $b = (c_{11}, \ldots, c_{1n}, c_{21}, \ldots, c_{2n}, \ldots, c_{n1}, \ldots, c_{nn})^T$, then the dual assignment problem (14) can be written as (1) with $A = (M\ E)$, where M is a block diagonal matrix with $M = \text{diag}\{e, e, \ldots, e\}$ and $E = (I, I, \ldots, I)^T$, in which $e = (1, 1, \ldots, 1)^T \in \mathbb{R}^n$ and I is the n-dimensional identity matrix. Then the proposed neural network in (4) is capable of solving the dual assignment problem.

Furthermore, for the dual assignment problem (14), the neural network in (4) can be written as the following component form: for $i = 1, 2, \ldots, n$

$$\begin{cases} \epsilon \dfrac{du_i}{dt} = -\sigma \sum_{j=1}^{n} g(u_i + v_j - c_{ij}) - 1, \\ \epsilon \dfrac{dv_i}{dt} = -\sigma \sum_{j=1}^{n} g(u_j + v_i - c_{ji}) - 1. \end{cases} \tag{15}$$

The solution from the dual assignment problem can be easily decoded into that for the primal assignment problem by using the complementary slackness theorem as follows:

$$x_{ij} = h(u_i + v_j - c_{ij}), \tag{16}$$

where $h(z)$ is the output function defined as $h(z) = 1$ if $z = 0$, or $h(z) = 0$ otherwise.

The neural network in (15) has one-layer structure only with $2n$ neurons (same as the number of decision variables in the dual assignment problem (14)). Compared with the primal neural network [4] with n^2 neurons and the primal-dual neural network [11] with $n^2 + 2n$ neurons, the proposed neural network herein has lower model complexity with one order fewer neurons. The proposed neural network has the same model complexity as the dual neural network in [4]. Nevertheless, the parameter selections for the dual neural network therein are not straightforward, whereas the proposed dual neural network herein is guaranteed to converge to exact optimal solutions as long as its single gain parameter in the model is larger than a derived lower bound.

According to the results in Section 3, the proposed neural network is capable of solving the dual linear assignment problem as in the following result.

Corollary 1. *The state vector of neural network* (15) *is globally convergent to an optimal solution of the dual assignment problem in finite time if* $\sigma > \sqrt{n}$.

Proof: As $c = (-1, -1, \ldots, -1)^T \in \mathbb{R}^{2n}$, $\|c\|_2 = \sqrt{2n}$. From the definition of A for the dual assignment problem (14), its elements can take 0 or 1 only. For any $\gamma \in \Gamma$, $\|A^T\gamma\|_2$ gets the minimum value if one element of γ is 1 and the others are 0, where Γ is defined in Section 3. By simple computation, for any $\gamma \in \Gamma$, we have $\|A^T\gamma\|_2 \geq \sqrt{2}$. Then, from (8), this corollary holds. □

5 Simulation Results

We consider an linear assignment problem with $n = 10$ and

$$[c_{ij}] = \begin{pmatrix} 1.6 & 6.7 & -3.6 & 2.8 & 5.6 & -4.1 & 10.5 & 4.2 & -2.8 & 10.9 \\ 1.6 & 9.3 & -5.8 & 5.1 & -4.6 & 10.2 & -1.2 & -2.0 & -2.2 & 8.3 \\ 8.0 & 11.8 & -5.2 & -1.7 & -2.4 & 10.4 & 3.9 & 9.1 & -7.8 & 10.6 \\ 1.4 & 2.1 & 1.2 & -8.4 & 3.2 & -7.7 & 4.3 & -5.7 & 3.5 & -1.8 \\ -4.2 & 4.6 & 7.5 & 0.5 & 1.7 & 7.4 & -3.9 & -2.2 & -5.0 & -2.6 \\ -3.6 & -6.2 & -2.4 & -2.1 & -5.0 & -4.9 & -3.6 & -6.0 & -2.1 & -2.7 \\ 5.3 & 1.0 & -3.5 & 5.5 & 3.4 & 8.3 & 9.1 & 0.5 & 4.6 & -2.7 \\ 5.5 & 2.5 & 10.2 & 10.8 & 1.6 & 8.5 & -5.8 & -4.3 & -2.9 & -3.8 \\ 10.9 & -4.6 & -7.8 & -1.2 & 2.5 & -4.0 & 0.8 & 8.9 & -5.1 & -3.6 \\ 7.4 & -6.6 & 3.7 & 3.3 & 7.8 & 5.5 & 6.0 & 8.7 & 5.1 & -1.2 \end{pmatrix}.$$

According to Corollary 1, a lower bound of σ is $\sqrt{10} \approx 3.1623$. Fig. 1 depicts the convergence of the state variables $(u(t), v(t))$ with $\epsilon = 10^{-5}$ and $\sigma = 3.2$. We can see that the state variables of neural network (15) are convergent to an optimal solution of the dual assignment problem (14) in finite time from a random initial point. Fig. 2 shows the convergence of the dual objective function of problem (14) (i.e., $\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}$ in the primal assignment problem) with three different values of σ. Using (16), the optimal solution to the corresponding primal assignment problem can be easily interpreted as follows:

$$[x_{ij}] = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

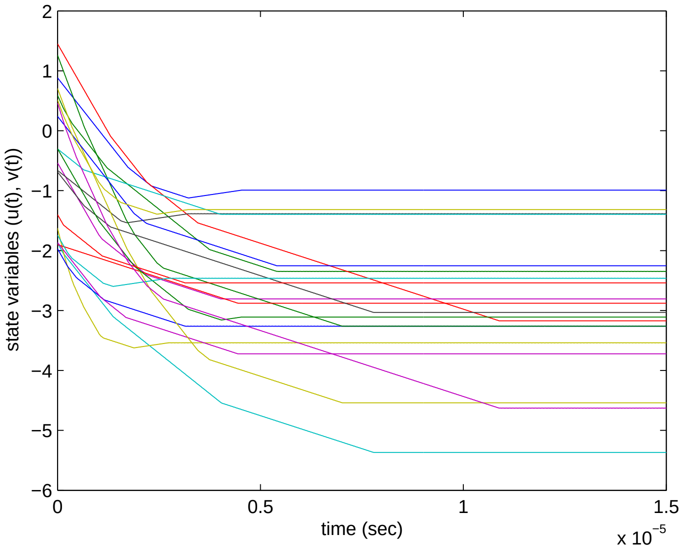


Fig. 1. Transient behaviors of the state variables of neural network (15) in the Example.

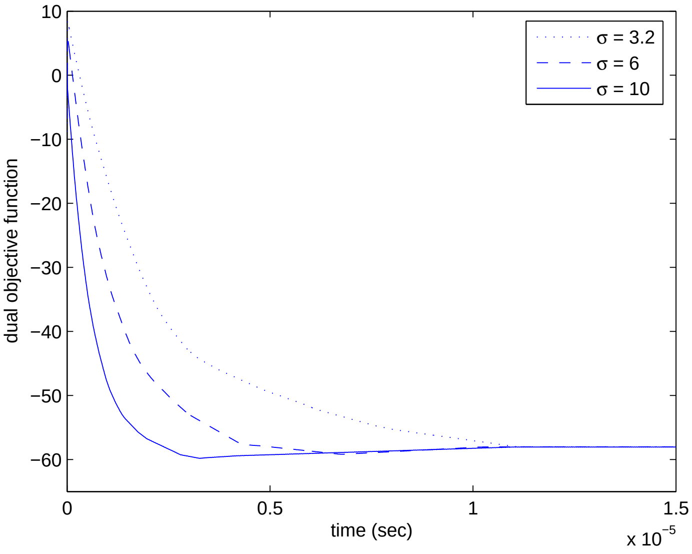


Fig. 2. Global convergence of the dual objective function of neural network (15) in the Example.

6 Conclusions

This paper presents a recurrent neural network with a unipolar hard-limiting activation function for solving linear programming problems. The finite-time global convergence of the proposed neural network to optimal solutions is guaranteed if its single gain parameter σ is larger than a derived lower bound. Furthermore, the proposed neural network is tailored for solving the linear assignment problems. Simulation results are given with a numerical example to illustrate the effectiveness and characteristics of the proposed neural network.

References

1. Tank, D., Hopfield, J.: Simple neural optimization networks: An a/d converter, signal decision circuit, and a linear programming circuit. IEEE Transactions on Circuits and Systems **33** (1986) 533–541
2. Kennedy, M., Chua, L.: Neural networks for nonlinear programming. IEEE Transactions on Circuits and Systems **35** (1988) 554–562
3. Wang, J.: Analysis and design of a recurrent neural network for linear programming. IEEE Transactions on Circuits and Systems-I **40** (1993) 613–618
4. Wang, J.: Primal and dual assignment networks. IEEE Transactions on Neural Networks **8** (1997) 784–790
5. Wang, J.: Primal and dual neural networks for shortest-path routing. IEEE Transactions on Systems, Man, and Cybernetics-A **28** (1998) 864–869
6. Xia, Y.: A new neural network for solving linear programming problems and its application. IEEE Transactions on Neural Networks **7** (1996) 525–529
7. Forti, M., Nistri, P., Quincampoix, M.: Generalized neural network for nonsmooth nonlinear programming problems. IEEE Transactions on Circuits and Systems-I **51** (2004) 1741–1754
8. Liu, Q., Wang, J.: A one-layer recurrent neural network with a discontinuous activation function for linear programming. Neural Computation **20** (2008) 1366–1383
9. Wang, J., Xia, Y.: Analysis and design of primal-dual assignment networks. IEEE Transactions on Neural Networks **9** (1998) 183–194
10. Wang, J.: A recurrent neural network for solving the shortest path problem. IEEE Transactions on Circuits and Systems-I **43** (1996) 482–486
11. Xia, Y., Wang, J.: A discrete-time recurrent neural network for shortest-path routing. IEEE Transactions on Automatic Control **45** (2000) 2129–2134
12. Bazaraa, M., Sherali, H., Shetty, C.: Nonlinear Programming: Theory and Algorithms (3rd Ed.). Hoboken, New Jersey: John Wiley & Sons (2006)
13. Liu, Q., Wang, J.: A one-layer recurrent neural network for convex programming. In: Proc. IEEE International Joint Conference on Neural Networks. (2008) 83–90
14. Walsh, G.: An Introduction to Linear Programming (2nd Ed.). Chichester: John Wiley & Sons (1985)

Towards a Holistic Approach addressing the Energy/Performance Tradeoff in Multi-Core Systems

Jan Richling[1], Jan Hendrik Schönherr[1], Matthias Werner[2], Gero Mühl[3]

[1] Fachgebiet Kommunikations- und Betriebssysteme, TU Berlin,
{richling|schnhrr}@cs.tu-berlin.de
[2] Professur für Betriebssysteme, TU Chemnitz, mwerner@cs.tu-chemnitz.de
[3] Architecture of Application Systems, University of Rostock,
gero.muehl@uni-rostock.de

Abstract. Current multi-core systems are able to deliver high performance for parallel as well as single-threaded workloads. Unfortunately, this performance comes at the cost of a high power usage. If the full performance is not needed, multi-core systems are nowadays able to reduce their energy consumption considerably. However, employed power management techniques often fail to exploit their potential. This is partly caused by an improper consideration of the provided architectural features resulting in a lower performance to energy ratio than necessary, and partly because of its appliance in unsuitable situations reducing users' acceptance. This paper sketches our vision of a holistic power management approach that respects users' needs, and we identify research challenges on the way to a fulfillment of this vision.

Keywords: Power management, Multi-core, Scheduling, Holistic approach

1 Current Situation

Only a few years ago, power management was mainly an issue for portable computers, where it directly affected mobility in terms of battery weight and run-time. However, due to increasing energy costs and increasing environmental awareness, the situation has changed fundamentally: Today, power management is considered essential for all areas of computing ranging from small sensors nodes over portable devices to large server farms. Regarding the desktop and server market, this is also because the power envelope of processors raised from only a few (e. g., 2 Watts for Intel's 80386) to as much as 140 Watts (e. g., AMD Phenom II 965). Therefore, methods to reduce energy consumption originally invented for mobile processors are now omnipresent. They apply mainly when a processor is not fully utilized or idle. Examples of these methods are voltage and frequency scaling (ACPI P-states[4]), which are used to adapt the computing

[4] For an overview on ACPI states see, e. g., [5].

power of a processor to the current load, as well as sleep states (ACPI C-states). These approaches are mostly beneficial for desktop systems as the processor often idles in such systems due to characteristics of interactive usage.

Today's HPC architectures use the very same off-the-shelf processors as on-the-edge servers and workstations, and thus, they feature the same power management properties. For HPC computers and server machines, the goal is usually to fully utilize the machines and therefore avoid idling at all eliminating any benefits from such approaches to power management. On the other hand, it is not always possible to fully utilize such a machine: Load may be lower than expected (especially in cases where a system is designed to guarantee given performance parameters) or a parallel program executed on a large HPC system has phases with low degree of parallelism forcing at least parts of the machine to idle. In such cases, also HPC machines benefit from this kind of power management.

With the advent of multi-core processors, power management features can be applied at the level of individual cores and additionally at the whole processor. For instance, cores may be driven into different P-states (if supported by hardware) and there may be a deeper sleep state for the whole processor which is entered when all cores are idle. Even more possibilities arise in multi-socket systems, where power management can be applied to each processor separately as well as for each core of each processor.

However, there is a pitfall when using all these power management features: Energy consumption is a non-functional property that must be considered in a holistic way, i. e., at *all* system layers. It is, thus, not enough to deal with it at the hardware layer and ignore it at upper layers such as the operating system and the applications. For example, an operating system which uses busy waiting to access a device may inhibit the processor from entering its deeper sleep states. Hence, operating systems and also application software must accompany the hardware means to reduce energy consumption. Moreover, not only features of the hardware (including processors) must be taken into account, but also characteristics of applications, usage scenarios, as well as user expectations. For instance:

- How does an application scale with increasing core frequency?
- Is it acceptable to execute a HPC application one third faster at doubled energy costs?
- Has the user an interest in maximizing the performance of a certain application?

Thus, only with this *combined* knowledge it is possible to balance energy consumption and performance optimally. This is not done in current systems and, even worse, functionality that is interdependent of power management is implemented without explicitly considering this dependency. This mainly concerns the scheduler, which decides when and where tasks are executed, since at the present time schedulers do not consider the current P-state (i. e., actual frequency) of a core when assigning tasks. In case it assigns a task to a processor core running at a low frequency, this leads to a performance degradation if this task would actually require a higher frequency to run best. The same is true if a task is running at a high frequency, although it would run with the same performance at a lower one, resulting in a higher energy consumption. These effects occur because, first, solely load is taken into account for deriving an appropriate frequency and, second, frequency adaptations are always lagging behind as they are usually done after periodical polls of the load. Furthermore, there is also a delay caused by switching the frequency and voltage in hardware. As we showed in a previous paper ([6]), these grievances together lead to the current situation where – in many cases – power management degrades performance so heavily that it is *completely* disabled by many users. Hence any potential to save energy, especially in situations where it would work effectively, is inevitably lost.

There is a further aspect that remains to be mentioned: each processor generation improves existing power management features and also introduces new ones to further reduce the power consumption of its components. Usually, operating system support comes *after* the appropriate hardware solution and older operating system solutions may not work or are even counterproductive on more advanced hardware. A good example for this are P-states which enable dynamic adjustment of the voltage and frequency of a processor to the actual demand. The idea is that due to the non-linear relation of power consumption and clock frequency, it is usually much more efficient to run tasks at lower frequencies if there is only light load, and, second, to reduce the frequency to the minimum if the processor core is idle. Although the first is true in most cases, C-states of modern processors are very effective by disabling large parts of a CPU up to complete cores when they are not needed. This way, it makes nearly no difference in energy consumption if an idle core runs at high or low frequency because it is almost completely disabled anyway. In essence, P-states are now of minor importance to save energy when idling, instead they are more relevant to run applications at their most energy-efficient frequency in case of light load.

For all these reasons, in this paper we depict our vision of what has to be done to balance energy consumption and performance in a holistic way. We believe that such an approach provides not only better results because it can avoid negative synergy effects, but it would also gain a better user acceptance, and thus, a better usage rate. We discuss some examples of existing approaches that would be part of a holistic solution as well as possible research directions to make this overall vision real.

The remainder of this paper is organized as follows: We start with introducing our objectives in Section 2. Based on this, we discuss a number of challenges as well as initial approaches in Section 3. The paper concludes in Section 4.

2 Objectives

We propose an integrated approach to power management and scheduling which balances energy consumption and performance optimally according to the users' need. We believe that any acting against the users' interests is doomed to fail. We, thus, declare the following main objective:

Energy saving should not affect performance as experienced or desired by the user significantly unless the user requests this.

We believe that it is inevitable to consider the user's preferences for each application and for different situations to come to a solution that gets no longer disabled but widely accepted. In order to make the objective real, we have identified three design criteria:

1. **All possibilities** offered by a system to optimize performance and energy efficiency should be considered and used, especially by the operating system. In particular, this means that all energy saving and performance boosting features of processors in a system should be exploited *in a coordinated way* and that scheduling algorithms should consider power management explicitly.
2. We want to replace the time-driven approach used by today's operating systems to adjust the frequency to the load by an **event-driven approach**. Instead of periodically polling load and setting frequencies accordingly afterwards, there will be a direct reaction to the arrival and removal of load by changing to the most appropriate frequency for this load immediately.
3. It is necessary to consider the **applications' characteristics** to choose the best mode to run them. This includes not only choosing the best frequency but also to determine the optimal number of processors for an application at a certain time and which of the available cores should be used. To achieve this, applications and operating system must closely cooperate.

In the following section, we describe some first approaches to make the vision of uniting energy efficiency and performance a reality, and name the challenges that have to be tackled.

3 Challenges

A typical HPC application consists of a burst of parallel threads, where the degree of parallelism may heavily fluctuate during the run of an application instance. We focus on the execution of such applications on multi-core processors addressing the fact that the number of cores is growing targeting "many-core" processors in the near future. This includes processors consisting of equal cores as well as asymmetric multi-cores because most of our ideas can be applied in both cases: asymmetry caused by different hardware as well as asymmetry caused by using different P-states. Furthermore, we put special attention to multi-core processors that are not fully utilized as not all software is able to utilize the degree of parallelism offered by current, and, more important, upcoming multi-cores. In case of HPC or server machines, such a situation occurs when the actual load is lower than the intended (full) load due to limited application level parallelism or phases with low load.

Modern processors have very efficient C-states which can disable complete cores, parts of them, or large regions of caches when unused. This dramatically decreases the energy consumption of an idling processor down to a point where it becomes useless to reduce the frequency in order to save energy when idling. Moreover, manufacturers start to implement functionalities such as frequency adaption directly into hardware. For example, Intel's latest generation processors, code-named Nehalem, feature a power control unit consisting of more than a million transistors. It does not only control C-states including to disable cores by power gating, but also the "Intel Turbo Boost" technology, a feature that increases performance by dynamically increasing clock frequency if limits for temperature, current and frequency are not reached. Similar approaches to dynamically increase clock frequency of some cores can be found in IBM's Power7 and AMD's six-core Phenom II code-named Thuban. Such hardware approaches try to intelligently respond to the behavior of the software system. If this software system, on the other hand, tries to do the same by, e. g., switching P-states, we have two controllers that may interfere in unpredictable ways with each other decreasing performance and maybe also increasing energy consumption. Therefore, our idea on the software side is not to control everything, but to make sure that the power management capabilities of the hardware are able to work most efficient. We will illustrate this approach with three examples in Sections 3.1, 3.2 and 3.3. Furthermore, we will present more details on the event-driven approach of frequency control mentioned above in Section 3.4 and introduce ideas on how to extend it in Sections 3.5 to 3.7.

3.1 Load Concentration

If we consider a not fully utilized system, modern operating systems try to balance the load between available cores. This generic strategy is subject to some optimizations, e. g., latest Linux kernels are aware of the topology of a machine and try to utilize package[5] after package thus restricting the load to a subset of available packages. Such optimizations are useful from an energy perspective as they allow the remaining packages to reside in deep sleep states. However, they only work on multi-package machines, i. e., they do not work on usual desktop machines or HPC nodes with just one package. This balanced scheduling leads to a more or less equally distributed average utilization of cores. But, in case of low degree of parallelism and many short activity phases this means that each core switches between active and sleep state at a quite high frequency. If we compile a Linux kernel with a parallelism degree of one (`make -j1`, here used as a benchmark consisting of many small tasks with low parallelism degree) on a quad-core, we have on average 100 transitions from load to idle or vice versa per core and per second. Here, our idea is to concentrate the load on as few cores as possible as proposed by us in [6].

Table 1 shows the results of such an experiment executed on an Intel Core i7 920 with disabled "Turbo Boost" and disabled Hyperthreading. It can be seen that

[5] A package is hereby a processor die. Usually, this is equal to a "socket" in a multi-socket machine. But there are exceptions, e. g., Intel's Core 2 quad-cores feature two packages per socket.

# cores	Run-time	Power	A	B
1	57:51 min	119.6 W	115.3 Wh	16.0 Wh
2	57:48 min	120.0 W	115.7 Wh	16.5 Wh
3	57:57 min	120.5 W	116.3 Wh	16.9 Wh
4	57:34 min	120.6 W	115.7 Wh	16.8 Wh

Table 1. Energy consumption and run-time of `make -j1` kernel compilation for different number of active cores with "Turbo Boost" and Hyperthreading disabled, A: whole computer, B: energy above idle (103 W)

the performance is only slightly affected while the overall energy consumption (whole computer, column A) is slightly reduced in case of concentrating the load on one core. If we only consider the energy needed additionally to idle consumption (what we have to do to calculate the energy needed to execute a task on a computer that runs anyway and is not turned off after execution, column B), the saving is higher. The absolute savings are rather low – that is due to the fact that this processor features a very efficient power management due to power gating of unused cores that is able to respond very fast. Nevertheless, the savings show that the approach still helps to save energy even for such an efficient implementation on the hardware side.

Although such an approach helps to reduce the power consumption, further research has to be done in order to ensure that energy is really conserved. Concentrating load on less cores reduces the overall amount of cache and, in case of NUMA systems, the available memory bandwidth. If an application is very cache sensitive so that by concentrating it on one core (or on one package in case of a multi-socket system) the number of cache misses increases dramatically, the performance may be reduced down to a point where the increased run-time increases energy consumption despite reduced power drain. Depending on the workload, it may – in special cases – be more efficient to *spread* the load so that memory bandwidth and usable cache capacity are maximized. Automatically recognizing such situations and adapting the policy is a challenge that has to be tackled.

3.2 Utilizing "Turbo Boost"

If we repeat the experiment described in Section 3.1 with "Turbo Boost" enabled (see Table 2), beside being faster due to higher clock frequencies, we profit much more from concentrating the load. For our compilation with a parallelism degree of one, we perform fastest if we concentrate the load to one core. Moreover, we consume less energy no matter if we consider the consumption of the whole computer or just the additionally needed energy of an otherwise idling machine. Furthermore, for the first case we need exactly the same amount of energy as in the non-turbo case albeit being much faster. On the other hand, "Turbo Boost" increases the power drain significantly which leads to worse results (compared to disabled "Turbo Boost") if we only consider the energy needed additionally on an otherwise idling computer.

# cores	Run-time	Power	A	B
1	53:12 min	130.0 W	115.3 Wh	24.0 Wh
2	53:39 min	130.9 W	117.0 Wh	24.9 Wh
3	54:03 min	130.6 W	117.7 Wh	24.9 Wh
4	53:58 min	131.9 W	118.7 Wh	26.0 Wh

Table 2. Energy consumption and run-time of `make -j1` kernel compilation for different number of cores, "Turbo Boost" activated, Hyperthreading disabled, A: whole computer, B: energy above idle (103 W)

This leads to the idea of incorporating "Turbo Boost" into energy savings. While this may sound counterproductive as "Turbo Boost" is a performance enhancement at first sight, we see a large potential here. The main disadvantage of "Turbo Boost" is that its dynamic frequency increase comes along with increased voltage. As the power consumption raises with the square of the voltage, this means a large increase in energy usage for a smaller increase in performance. Therefore, it should be disabled for saving energy. On the other hand, as discussed at the beginning, such a "solution" is not acceptable for most users because of the performance penalty. Hence, we want to provide means to efficiently utilize "Turbo Boost" if profitable.

"Turbo Boost" increases clock frequency of one or more (up to all) cores if the thermal design power (TDP), the maximum temperature and a predefined limit on frequency are not yet reached. Therefore, it accelerates everything from unimportant background processes without any time constraints to important foreground applications. Here, we see the first possibility for improvement: The operating system disables "Turbo Boost" if only unimportant tasks are executed. This could be done be adding a "green" flag or by just using a mechanism such as Unix `nice` in a way that tasks with a positive nice-value are executed with disabled "Turbo Boost", or, even more effective, at their most energy-efficient frequency. Unfortunately, "Turbo Boost" is activated per package, not per core. Thus, this approach is only efficient if scheduling is adapted accordingly, so that such tasks are executed only if there are no important tasks at other cores of the same package at the same time. This may delay such tasks but on the other hand it efficiently avoids wasting of energy due to such unimportant activities.

The second option for improvement targets important tasks: The amount of performance increase due to increasing clock frequency depends on the number of active cores. This is defined by the manufacturer using a machine specific register (MSR) containing something like 1/1/1/2 meaning that for only one active core the frequency can be increased by two 133 MHz steps while 2 to 4 active cores may still get a boost of one step. The actual setting depends on the processor used. This way, a very important task may be slowed down if other (unimportant) tasks utilize other cores at the same time resulting in lower turbo frequencies. Therefore, a mechanism similar to the first seems to be useful: If we avoid to utilize more than one core of the processor if one core is used by an important task, this task gets the maximum boost. Especially on newer Nehalem processors this is significant, e. g., the Xeon L3426 has a 2/2/9/10 configuration so the step from 2 to 3 active cores may lead to a reduction in clock frequency by 933 MHz. In combination with separation of important and unimportant tasks this idea might also reduce overall energy consumption because first executing the important task isolated with maximum turbo frequency and later executing the unimportant tasks at their most energy-efficient frequency might need less energy than executing both together at reduced turbo frequency.

In [8], we investigated these ideas using a mixed workload scenario consisting of a varying number of important foreground and unimportant background tasks. We figured out (Figure 1) that by coarse-grained adaption of scheduling according to the ideas described above important tasks always get a performance boost (compared to the reference case of just enabling "Turbo Boost" without changing scheduling) while the overall energy consumption (caused by important and unimportant tasks) is reduced in a number of cases. Furthermore, in other cases the performance boost comes at the cost of a minor increase in energy consumption (still reducing the energy-delay product). We observed only few cases where the energy consumption is dramatically increased for a small gain in performance. The first goal for future research is therefore to automatically identify the behavior of the workload and apply the approach only if it is able to both increase performance and decrease overall energy consumption for such mixed workloads (or, depending on the preference of the user, to accept a small increase in energy consumption for a large performance boost). The second goal for future improvements is to replace the coarse-grained control used in [8] by a fine-grained control that incorporates the approach into scheduling. A promising idea is to use gang scheduling ([4]) in order to isolate execution of important and unimportant tasks from each other allowing to use turbo frequencies for the first and the most efficient frequency (as described in Section 3.5) for the latter.

3.3 Utilizing Hyperthreading

Another interesting hardware feature offered by Intel's Nehalem is its implementation of SMT ([3]) called Hyperthreading. By efficiently utilizing the resources of physical cores, two threads can be executed in parallel on each core. The performance boost depends on the actual code executed. From an energy perspective, using Hyperthreading increases the power consumption but, more important, also increases efficiency. Nevertheless, SMT-aware schedulers try to first utilize physical cores and then logical cores ([9]). In scenarios with low parallelism degree this may lead to situations where two physical cores are used in cases where both threads of one physical core would be sufficient (especially in combination with "Turbo Boost"). In this case, an additional core could go into a sleep state conserving more energy. The challenge here is to dynamically recognize such situations and to adjust the load concentration described in Section 3.1 so that it profits from this idea. However, there may also be cases in which Hyperthreading leads to a performance degradation. This can, e. g., arise if the L1/L2 caches of a core are too small to execute both current threads without increasing the

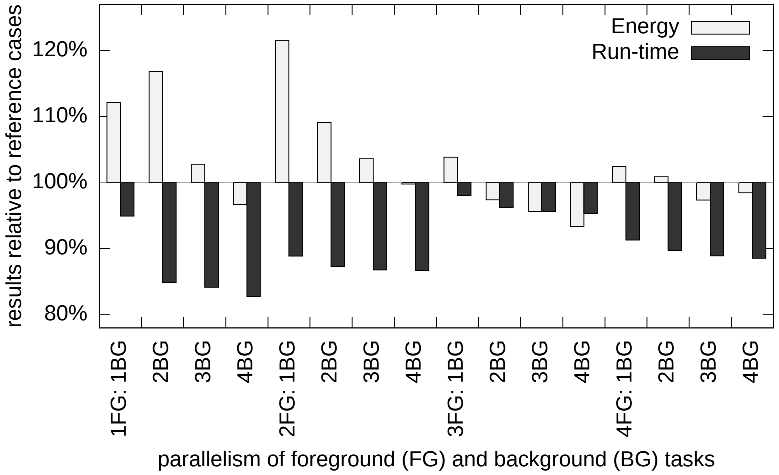


Fig. 1. Run-times of foreground load and overall energy consumption relative to the corresponding reference cases.

cache miss rate. The appearance of these cases have to be detected and properly handled.

3.4 Event-driven Frequency Control

One of the major design criteria described in Section 2 is to replace the traditional time-driven way to estimate actual load of the system by an event-driven approach. Instead of using one component of the kernel to periodically detect the actual load and adjusting the frequency accordingly, it makes much more sense to respond to load changes immediately by coupling the decisions on power management transitions to events that introduce or remove load. These events are available inside the scheduler (state changes of tasks) so an event-driven power management governor has to be part of the scheduler. In [7], we presented a first implementation of this approach. Figure 2 shows the performance impact of the approach (governor “scheduler”) in comparison to other energy governors while Table 3 presents the measurements of energy consumption. It can be seen that the approach is promising in order to conserve energy with nearly no performance penalty. While the used experiment had a large number of short running processes inducing many load changes, the event-triggered approach is also promising for long running processes as it reduces the number of executions of the governor to two (compared to 10 to 100 per second using the time-triggered approach): One at the moment the load occurs and the second after it finishes execution. Next steps of research with respect to this approach are described in the following two sections.

Governor	Run-time	Power	Energy	EDP
Performance	1:13:39 h	132 W	162 Wh	199 Whh
Scheduler	1:14:35 h	120 W	149 Wh	185 Whh
Ondemand	1:19:43 h	117 W	156 Wh	207 Whh
Powersave	2:18:43 h	107 W	246 Wh	569 Whh

Table 3. Energy consumption of compiling a Linux kernel with `make -j1` for different governors on AMD Phenom 9950 (EDP: energy-delay product).

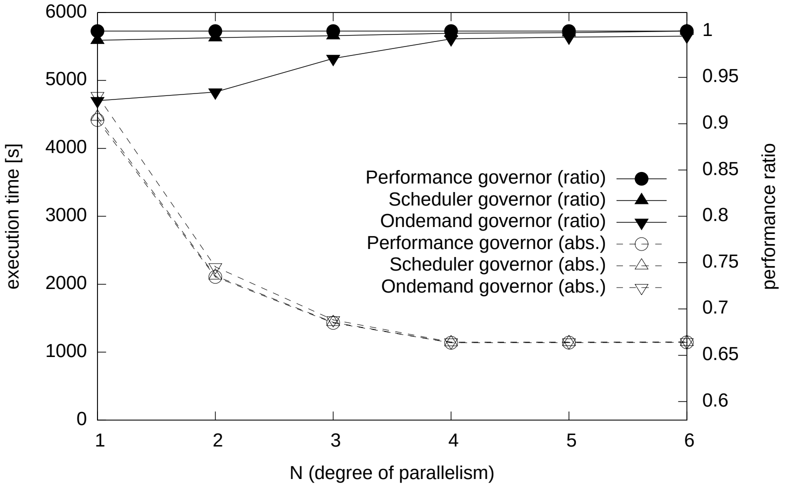


Fig. 2. Execution times and performance ratios of compiling a Linux kernel with `make -j`N on AMD Phenom 9950.

3.5 Optimizing Energy-Efficiency

In the real world, the execution performance of tasks is limited by several factors. Some tasks depend on processing power of the CPU only (called *CPU-bound*) and, therefore, profit directly from higher clock frequencies. Others perform a lot of memory accesses and are, thus, limited mainly by the speed of the memory subsystem. Beyond a certain point such *memory-bound* tasks do not profit from increased clock frequencies. The same is true for *I/O-bound* tasks that are limited by the performance of some I/O device they are using, e. g., communication hardware such as SCI or Infiniband in case of a HPC system. Depending on the ratio between computation and memory accesses on the one hand and the ratio between communication and computation on the other hand, HPC applications can be of any of those three types.

For CPU-bound tasks our goal of reducing energy consumption without loosing performance implies that the highest clock frequency is always required if it is a performance critical task. On the other hand, tasks bound by memory or I/O do not profit from such an increase. Therefore, running them at the highest frequency just wastes energy without any performance gain. Here, we need to derive the optimal operating frequency, e. g., by using an approach as described in [10].

Beside this, the combination with load concentration as described in Section 3.1 is promising: If we consider a memory-bound multi-threaded application executed on a multi-core processor with a memory controller shared between cores, it makes sense to concentrate the application on as many cores as needed to saturate the available memory bandwidth, thereby restricting its number of simultaneously active threads. Allowing more cores only wastes energy without any benefit. Such an approach is especially effective if the application interacts with the operating system in order to adjust the degree of parallelism. Furthermore, this idea can also be applied to workloads composed of memory-bound threads from different applications.

The challenge that we have to tackle here is the event-driven estimation of the current application behavior, i. e., without periodic calculations based on observations gathered during the last period. A second challenge is to interact with hardware control such as Nehalem's power control unit so that our decisions cooperate nicely with the decisions taken by hardware and do not result in a situation where a hardware controller and a software controller work against each other.

3.6 Combining C-state and P-state Control

As already stated, current processors feature sophisticated C-states. Therefore, the operating system has to use these C-states in a way that cores are able to stay in these (non C0) states as long as possible and enter them as early as possible. The load concentration described in Section 3.1 is a promising way to do this. Additionally, these sophisticated C-states make it obsolete to reduce clock frequencies in case of idling. Instead, doing so as done currently by most governors wastes energy as cores leave their C-state just to switch P-states. Therefore, it is essential to combine P-state and C-state control in one component (instead of using two separate components as done in, e. g., Linux). We propose to do that using the event-driven approach described in Section 3.4. That way, P-states are only used to ensure energy efficiency as described in Section 3.5 while C-states are entered in case of idling cores. Furthermore, such a combined approach allows to incorporate properties of the underlying hardware: For example, legacy systems with less efficient C-states still rely on frequency control to additionally enter the lowest P-state when idling to maximize energy savings. Using the combined approach, this can be done in a coordinated way.

3.7 Application Development

Energy is a non-functional property that has to be addressed an *all* layers including the applications. In the following, we discuss two consequences with respect to applications and their development.

First, we believe that in future the design of HPC applications must change. While applications might still use dedicated hardware, this hardware must be used differently as it is not homogeneous anymore. Today, hardware features to maximize speed or throughput get disabled (e. g., [2] and [1]) to get “repeatable” or more “predictable” results. The same holds for features to minimize energy consumption: frequency control gets disabled, processes get bound to certain processors. Contrary to that, we believe these features are not a burden, they are a chance. In the future, applications should not work against the operating system and thereby prevent energy savings. Furthermore, applications must deal with a certain performance imbalance within a computer system: inevitably some threads will finish their work earlier, others later.

Second, we believe that energy awareness should not be handled by a set of individual control mechanisms but, instead, by one cross-layer entity that incorporates all individual approaches and additionally addresses the relations and dependencies between them resulting in a holistic energy awareness. Doing so, the design principle of the single point of truth can be realized, avoiding inconsistent decisions of competing entities that may lead to ineffective or even harmful effects. However, a crucial point is the gain of information that is used to make a decision. While the discussed event-based power management provides more time-accurate data than cyclic polling, there is still information that is not accessible at this level. For this reason, we suggest the use of metadata provided by the compiler or other tools of the development tool chain, e. g., to get the maximal effect from gang scheduling as proposed in Section 3.2, the compiler may give hints for proper configuration. Even more, the other way around has to be considered also: Development tools could be made aware of the applied principles of a holistic energy-aware scheduling, thus they can reduce the probability of performance reductions due to cache misses as discussed in Section 3.1. However, all this measures should be seen as additional pieces in a common, holistic effort towards energy efficiency. It has to be assured that none of them invalidates the user orientation as stated as the overall objective in Section 2.

4 Conclusion

In this paper, we presented our vision on how to save energy without degrading performance experienced by the user by applying holistic approaches. We introduced a set of ideas to efficiently support power saving features of modern processors within the operating system, as well as approaches to save energy on multi-core systems with low to medium utilization. For all approaches, we identified research challenges that have to be solved to make these approaches feasible.

As next steps, we plan to tackle every single one of the described challenges and to integrate the solutions into a holistic approach that considers all introduced interdependencies. It is our hope to provide an approach not only with a better user acceptance, but also with a higher efficiency than a pure combination of separately designed measures can provide.

Furthermore, it is our believe that holistic approaches are a necessary step towards energy-aware systems as the cross-layer nature of the problem cannot be addressed using separate solutions at individual layers.

References

1. Kevin J. Barker, Kei Davis, Adolfy Hoisie, Darren J. Kerbyson, Mike Lang, Scott Pakin, and José Carlos Sancho. A performance evaluation of the Nehalem quad-core processor for scientific computing. *Parallel Processing Letters*, 18(4):453–469, 2008.
2. Alexandre Borghi, Jerome Darbon, Sylvain Peyronnet, Tony F. Chan, and Stanley Osher. A simple compressive sensing algorithm for parallel many-core architectures. Computational and Applied Mathematics Reports 08-64, University of California, Los Angeles, USA, September 2008. Revised August 2009.
3. Jeff Casazza. *Intel Core i7-800 Processor Series and the Intel Core i5-700 Processor Series Based on Intel Microarchitecture (Nehalem), Whitepaper*. Intel Corporation, Santa Clara, CA, USA, 2009.
4. Dror G. Feitelson and Larry Rudolph. Gang scheduling performance benefits for fine-grain synchronization. *Journal of Parallel and Distributed Computing*, 16:306–318, 1992.
5. Hewlett-Packard Corporation, Intel Corporation, Microsoft Corporation, Phoenix Technologies Ltd., and Toshiba Corporation. Advanced Configuration and Power Interface specification (ACPI), revision 4.0, June 2009.
6. Jan Richling, Jan H. Schönherr, Gero Mühl, and Matthias Werner. Towards energy-aware multi-core scheduling. *Praxis der Informationsverarbeitung und Kommunikation (PIK)*, 32(2):88–95, April-June 2009.
7. Jan H. Schönherr, Jan Richling, Matthias Werner, and Gero Mühl. Event-driven processor power management. In *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, pages 61–70, New York, NY, USA, April 2010. ACM Press.
8. Jan H. Schönherr, Jan Richling, Matthias Werner, and Gero Mühl. A scheduling approach for efficient utilization of hardware-driven frequency scaling. In *Workshop Proceedings of the 23rd International Conference on Architecture of Computing Systems (ARCS '10)*, pages 367–376, Berlin, Germany, February 2010. VDE Verlag.
9. Suresh Siddha, Venkatesh Pallipadi, and Asit Mallick. Chip multi processing aware linux kernel scheduler. In *Proceedings of the Linux Symposium*, volume 2, pages 329–340, July 2006.
10. Andreas Weissel and Frank Bellosa. Process Cruise Control: event-driven clock scaling for dynamic power management. In *Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems (CASES '02)*, pages 238–246, New York, NY, USA, October 2002. ACM Press.

Constrained Nonnegative Tensor Factorization for Speech Emotion Recognition

Guangchuan Shi[1], Qiang Wu[1], Liqing Zhang[1]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University,
Shanghai, China
[1]MOE-Microsoft Key Laboratory for Intelligent Computing and Intelligent Systems
{sgc1984,johnnywu}@sjtu.edu.cn,zhang-lq@cs.sjtu.edu.cn

Abstract. This paper presents an experimental study of the feature extraction framework based on constrained nonnegative tensor factorization using Gabor analysis for speech emotion recognition. Gaussian mixture models are used to modeling the distribution of different emotions in the feature space. Experiments are performed on FAU Aibo Emotion Corpus. Recognition results show that our proposed feature extraction framework achieves 49.13% weighted accuracy which gains 8.24% increase comparing with the MFCC based feature extraction method.

Key words: constrained nonnegative tensor factorization, Gabor filtering, Gaussian mixture model, emotion recognition

1 Introduction

Automatic emotion recognition (AER) for speech signal has attracted more attention in recent years ([1]). Emotion recognition is a procedure that recognizes the human emotion symbols from the speech signal, e.g. anger, excitation, sadness, happiness or neutrality. It plays an important role in *natural language processing* (NLP) especially as an important module in *human-computer interface* (HCI). By AER, HCI system can understand not only what the verbal content of human says but also the underlying emotion in the speech and get more accurate understanding of human during the interaction.

A typical AER system usually includes two modules: feature extraction and pattern recognition. For pattern recognition, common modeling methods such as hidden Markov model (HMM), support vector machine (SVM), Gaussian mixture model (GMM), k-nearest neighbors (kNN) have been proved efficient for emotion recognition ([2], [3], [4], [5]). We can find that the difference between different pattern recognition methods is rather small, so the feature extraction method maybe acts an important role in the AER system. For feature extraction, prosodic and statistical features of speech such as formant frequency, zero crossing rate, log energy, root mean square and etc. are commonly used ([6], [2], [3]). Besides, high dimensional Mel-frequency cepstral coefficient (MFCC) gains a significant improvement in some applications ([7]). But the recognition rate is low while the speakers are different in training and testing stage, especially different gender or different accent.

Tensor, as a generation of matrix, is a powerful framework for data analysis. Motivated by the nonnegative matrix factorization (NMF), tensor decomposition models have been proposed including CANDECOMP/PARAFAC model ([8]), Tucker model ([9]) and nonnegative tensor factorization (NTF, [10]). Not only in speech community but also in other intelligent information processing fields, many applications show that tensor representation can preserve more structural information of the data than matrix representation and provide an outstanding performance improvement.

Gabor filtering is commonly used in computer vision to modeling the primary visual cortex (V1). Neuronphysiological evidence indicates that the primary auditory cortex (A1) has the similar spectro-temporal response fields (STRF) with the primary visual cortex. So many attempts of using Gabor filtering for speech applications have been performed ([11], [12], [13], [14]). It has been proved a significant improvement of feature robustness for speech recognition and speaker recognition in noisy environment ([13], [14]).

In this paper, we investigate the performance of emotion recognition system using Gabor filtering based on the tensor structure. We employ the Gabor filtering to analyze the auditory speech spectrum and project features into a new subspace using the basis matrices estimated by the constrained nonnegative tensor factorization (cNTF) algorithm ([14]). GMM codebooks are trained for each emotion label by several EM iterations. During the recognition procedure, we estimate the posteriors of all the GMM codebooks and the emotion label whose GMM has the maximal probability is labeled for the testing utterance. Neuronphysiological experiments show that there are only a few neuron cells that response to a certain stimulus. Sparse coding theory gives us the way to control how many neuron cells activated by the stimulus. Orthogonal constraint of the basis matrices can de-correlate the linear correlation between different dimensions of the feature, and obtain a more accurate estimation of the distribution of the feature space using GMM with diagonal covariance matrix. cNTF algorithm imposes these two constraints to the original NTF algorithm and achieves a significant improvement on the recognition accuracy in our experiment.

The remainder of this paper is organized as follows. In Section 2, we will introduce our speech emotion recognition system, and the experimental results will be illustrated in Section 3. Finally, we will give a conclusion of this paper in Section 4.

2 Method

2.1 System Overview

Figure 1 demonstrates the overview of our proposed AER system. Upper part is the training procedure. Power spectrum is generated by short-time Fourier transform (STFT) from the training data, then we use the Gabor filter to analyze the spectrum and the cNTF is performed to get the basis matrices. Besides, we obtain the GMM codebooks for each emotion by the GMM training module. Lower part is the recognition procedure. For each testing utterance, acoustic feature is extracted by the same procedure as in the GMM training procedure. Then for each frame of the utterance, we choose the emotion label whose GMM codebook achieves the maximal posterior probability. Finally, according to the emotion labels of each frame, the voting strategy determines the emotion label for the whole utterance.

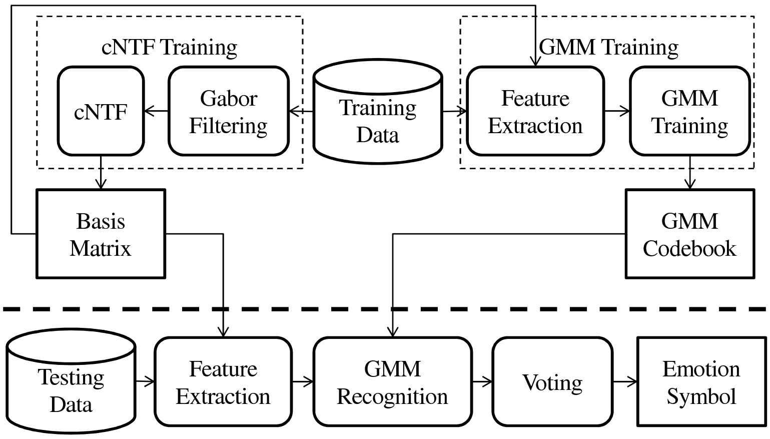


Fig. 1. System Overview

2.2 Constrained Nonnegative Tensor Factorization

A tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_M}$ is a multidimensional array where M is the order. It is a generation of the matrix which can be treated as a 2-order tensor. In PARAFAC model, an M-order tensor can be represented by a sum of M-order rank-1 terms:

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{A}_r^{(1)} \circ \mathbf{A}_r^{(2)} \circ \cdots \circ \mathbf{A}_r^{(M)}, \tag{1}$$

where $\mathbf{A}_r^{(d)}$ is the r-th column vector of the mode-d basis matrix $\mathbf{A}^{(d)} \in \mathbb{R}^{N_d \times R}$, R is the rank of the tensor $\mathcal{X}$ which is the minimal number of the rank-1 terms and $\circ$ is the vector outer product operator.

PARAFAC model aims to find a rank-R approximation as Eq (1) for the tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_M}$, i.e.,

$$\mathcal{X} \approx \hat{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{A}_r^{(1)} \circ \mathbf{A}_r^{(2)} \circ \cdots \circ \mathbf{A}_r^{(M)}, \tag{2}$$

or in elementwise representation

$$\mathcal{X}_{n_1,n_2,\ldots,n_M} \approx \hat{\mathcal{X}}_{n_1,n_2,\ldots,n_M} = \sum_{r=1}^{R} \mathbf{A}^{(1)}_{n_1,r} \circ \mathbf{A}^{(2)}_{n_2,r} \circ \cdots \circ \mathbf{A}^{(M)}_{n_M,r}. \tag{3}$$

As another representation for Eq (2), the PARAFAC approximation can be rewritten as:

$$\mathcal{X}_{(d)} \approx \mathbf{A}^{(d)} \left[\mathbf{A}^{(d-1)} \odot \cdots \mathbf{A}^{(1)} \odot \mathbf{A}^{(M)} \odot \cdots \odot \mathbf{A}^{(d+1)}\right]^T, \tag{4}$$

where $\odot$ is the Khatri-Rao product operator and $\mathcal{X}_{(d)} \in \mathbb{R}^{N_d \times \prod_{i \neq d} N_j}$ is mode-d matricization of tensor $\mathcal{X}$ ([14]).

Given a nonnegative tensor $\mathcal{X} \in \mathbb{R}_+^{N_1 \times N_2 \times \cdots \times N_M}$, NTF algorithm estimates the nonnegative basis matrices $\mathbf{A}^{(d)} \in \mathbb{R}_+^{N_d \times R}, d = 1, 2, \ldots, M$ by minimizing the KL-divergence cost function $\mathcal{F}_{KL}(\mathcal{X}, \hat{\mathcal{X}})$ as follows:

$$\begin{aligned}\mathcal{F}_{KL}(\mathcal{X}, \hat{\mathcal{X}}) &= \mathrm{KL}(\mathcal{X}||\hat{\mathcal{X}}) \\ &= \sum_{n_1,n_2,\ldots,n_M} \Bigg(\mathcal{X}_{n_1,n_2,\ldots,n_M} \log \frac{\mathcal{X}_{n_1,n_2,\ldots,n_M}}{\hat{\mathcal{X}}_{n_1,n_2,\ldots,n_M}} \\ &\quad -\mathcal{X}_{n_1,n_2,\ldots,n_M} + \hat{\mathcal{X}}_{n_1,n_2,\ldots,n_M}\Bigg). \end{aligned} \tag{5}$$

Using the mode-d matricization notation, Eq (5) can be rewritten as follows:

$$\begin{aligned}\mathcal{F}^{(1)}_{KL}(\mathbf{A}^{(d)}) &= \sum_{d=1}^{M} \mathrm{KL}(\mathcal{X}_{(d)}||\hat{\mathcal{X}}_{(d)}) \\ &= \sum_{d=1}^{M}\sum_{p=1}^{N_d}\sum_{q=1}^{\overline{N_d}} \Bigg([\mathcal{X}_{(d)}]_{pq} \log \frac{[\mathcal{X}_{(d)}]_{pq}}{[\mathbf{A}^{(d)}\mathbf{Z}^{(d)}]_{pq}} \\ &\quad -[\mathcal{X}_{(d)}]_{pq} + [\mathbf{A}^{(d)}\mathbf{Z}^{(d)}]_{pq}\Bigg), \end{aligned} \tag{6}$$

where $\mathbf{Z}^{(d)} = \left[\mathbf{A}^{(d-1)} \odot \cdots \mathbf{A}^{(1)} \odot \mathbf{A}^{(M)} \odot \cdots \odot \mathbf{A}^{(d+1)}\right]^T$ and $\overline{N_d} = \prod_{j=1,j\neq d}^{M} N_j$

By imposing the orthogonal penalty term and smoothing (sparse) matrix $\mathbf{S} \in \mathbb{R}^{R \times R}$ ([15]) to Eq (6), we get the constrained NTF cost function:

$$\begin{aligned}&\mathcal{F}^{(2)}_{KL}(\mathbf{A}^{(d)}) \\ &= \sum_{d=1}^{M} \Bigg[\sum_{p=1}^{N_d}\sum_{q=1}^{\overline{N_d}} \Bigg([\mathcal{X}_{(d)}]_{pq} \log \frac{[\mathcal{X}_{(d)}]_{pq}}{[\mathbf{A}^{(d)}\mathbf{S}\mathbf{Z}^{(d)}]_{pq}} \\ &\quad -[\mathcal{X}_{(d)}]_{pq} + [\mathbf{A}^{(d)}\mathbf{S}\mathbf{Z}^{(d)}]_{pq}\Bigg) + \alpha \sum_{p\neq q} [\mathbf{A}^{(d)T}\mathbf{A}^{(d)}]_{pq}\Bigg], \end{aligned} \tag{7}$$

where $\mathbf{S} = (1-\theta)\mathbf{I} + \frac{\theta}{R}\mathbf{1}\mathbf{1}^T$ is the smoothing matrix, $\mathbf{I}$ is identical matrix, $\mathbf{1}$ is a column vector of ones, $0 \leq \theta \leq 1$ is the smoothing factor and $\alpha \geq 0$ is the orthogonal factor.

By applying traditional exponential gradient iteration algorithm on Eq (7), cNTF algorithm estimates the basis matrices $\mathbf{A}^{(d)}$ (for more detail, please refer [14]).

2.3 Gabor Filtering

According to the physiological and psychoacoustic experimental results of auditory system, spectro-temporal response field (STRF) of the primary auditory cortex (A1) cell has been proposed to simulate the response of the neuron cell of A1 to the stimulus ([12]). This can be modeled by the 2D complex Gabor filtering ([11]) which is a product of the Gaussian envelope and a complex plant wave, i.e.,

$$\mathbf{G}_{u,v}(f,t) = \mathbf{G}_{\mathbf{k}}(\mathbf{x}) = \frac{\mathbf{k}^T\mathbf{k}}{\sigma^2} \cdot \exp\left\{-\frac{\mathbf{k}^T\mathbf{k}\cdot\mathbf{x}^T\mathbf{x}}{2\sigma^2}\right\} \cdot \left[\exp\{i\mathbf{k}^T\mathbf{x}\} - \exp\{-\frac{\sigma^2}{2}\}\right], \tag{8}$$

where vector $\mathbf{x} = [f,t]^T$ and $\mathbf{k} = [k_v \cos\phi, k_v \sin\phi]^T$, $\sigma = \sqrt{2}\pi$, $k_v = \pi \cdot 2^{-\frac{v+2}{2}}$, $\phi = \frac{u}{K}\pi$. u determines the orientation of the Gabor function while there are K directions totally in the filterbank and v controls the scale of the Gabor function.

2.4 Feature Extraction

In our feature extraction module, an utterance y with emotion label e is firstly pre-emphasized, Hamming window is applied to split the speech signal into frame and short-time Fourier transform (STFT) is performed to get the power spectrum $\mathbf{S}$. Then we convolute $\mathbf{S}$ using the Gabor filters $\mathbf{G}_{u,v}$ to obtain the STRF response of the spectrogram of different scales and orientations. Mel-frequency filterbank is used thereafter to obtain the auditory STRF response represented by a tensor $\mathcal{S} \in \mathbb{R}^{N_f \times N_t \times N_o \times N_s}$ where N_o and N_s is the number of the orientations and scales of the Gabor filter respectively, N_f is the number of bins of Mel-frequency filterbank and N_t is the number of the frames,

$$\mathcal{S}_{:,:,u,v} = \mathrm{Mel}\left\{|\mathbf{S} \otimes \mathbf{G}_{u,v}|\right\}, \tag{9}$$

where $\otimes$ is the convolution operator. Gathering the tensors $\mathcal{S}$ of all emotions, we construct the 5-order input tensor $\overline{\mathcal{S}} \in \mathbb{R}^{N_f \times N_t \times N_o \times N_s \times N_e}$ with the emotion label as the last mode index. The basis matrices $\mathbf{A}^{(d)}$ can be estimated by cNTF algorithm for the Gabor tensor $\overline{\mathcal{S}}$. On the other hand, we need the following steps to perform the feature extraction. Using the mode-1 basis matrix $\mathbf{A}^{(1)}$, we project the Gabor tensor $\mathcal{S}$ into $\mathcal{S}^*$ as follows:

$$\mathcal{S}^*_{:,:,u,v} = \mathbf{U} \times \mathcal{S}_{:,:,u,v}, \tag{10}$$

where $\mathbf{U}$ is the positive elements of the pseudo-inverse of $\mathbf{A}^{(1)}$, i.e., $\mathbf{U} = [\mathbf{A}^{(1)}]_{+}^{-1}$. Then, we unfold the tensor $\mathcal{S}^*$ by mode-2 tensor matricization, log and discrete cosine transform (DCT) operations are performed sequentially to convert into cepstral coefficients $\mathbf{C}$, i.e.,

$$\mathbf{C} = \mathrm{DCT}\left(\log \mathcal{S}^*_{(2)}\right) . \tag{11}$$

Finally, cepstral liftering is employed to balance the scale of different dimensions of the feature.

In summary, Figure 2 concludes the procedure of the feature extraction.

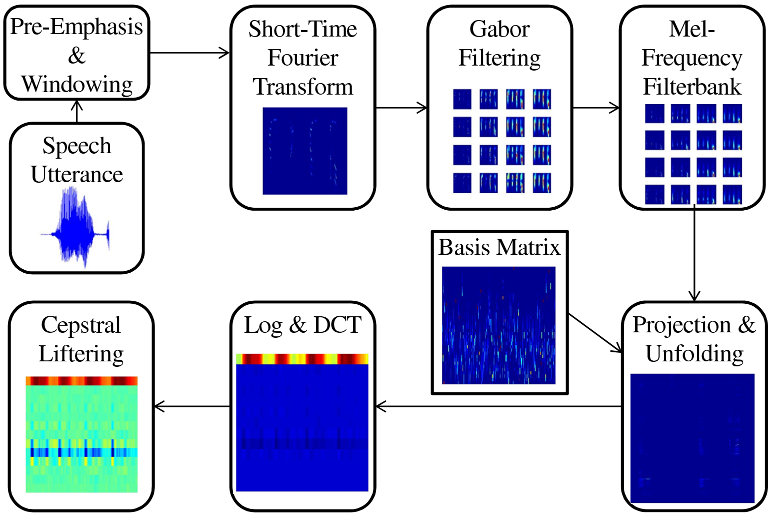


Fig. 2. An Overview of Feature Extraction

3 Experimental Results

We use the FAU Aibo Emotion Corpus ([16], [17]) for evaluating the performance of our feature extraction method for speech emotion recognition. 11 emotions of German speech are contained within 9 hours speech data totally which is spoken by 51 children at the age between 10 and 13 interacting with Sony's pet robot Aibo. In our experiment, we use 5 emotion classes of them including ***Anger*** (subsuming *angry*, *touchy* and *reprimanding*), ***Emphatic***, ***Neutral***, ***Positive*** (subsuming *motherese* and *joyful*) and ***Rest***.

Speech signals are down-sampled into 8KHz. A 25ms Hamming window with 10ms shifting is used for obtaining the power spectrum using STFT. 36 Mel-frequency filterbank is imposed in our experiment for estimating the auditory power spectrum. The Gabor filter is a set of 41 × 41 patches with 4 orientations and 4 scales. A 5-order tensor (frequency bin × time frame × Gabor orientation × Gabor scale × emotion label) is generated for estimating the basis matrices in cNTF using Gabor filtering with emotion label as the last mode index. 10 seconds speech data for each emotion is in use (i.e., 50 seconds in all). The rank of the tensor R is set to 200 and the orthogonal factor $\alpha = 0.005$ while smoothing factor $\theta = 0.05$. We reserve the first 13 DCT coefficients $(c_0, c_1, \ldots, c_{12})$ with delta (Δ) and acceleration $(\Delta\Delta)$ as the final 39-dimensional feature. For comparison, we build the baseline system by 13 MFCC feature $(c_0, c_1, \ldots, c_{12})$ with delta (Δ) and acceleration $(\Delta\Delta)$.

Gaussian mixture model is used for modeling the distribution of different emotions in the feature space. The GMM with 64 components is trained using 60 seconds speech data for each emotion by maximal 100 EM iterations.

4700 utterances which are randomly selected are recognized for evaluating the performance. For each utterance, recognition module assigns an emotion label for each frame of the utterance which has the maximal posterior probability. Then we set the emotion label for the whole utterance which has the most amount voted by emotion labels of all frames.

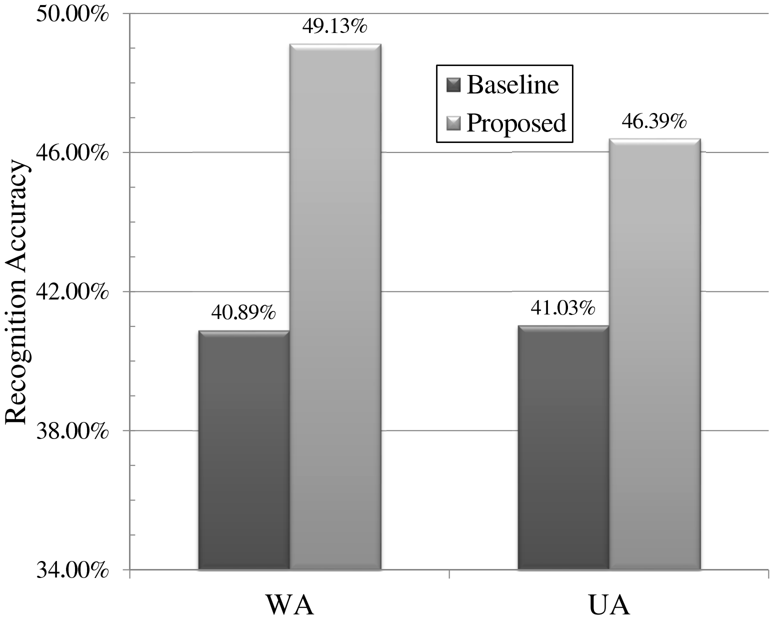


Fig. 3. Recognition Accuracy

Figure 3 demonstrates the weighted accuracy (WA) and unweighted accuracy (UA) using different feature extraction methods. Comparing the WA of the two methods, our proposed method gains 8.24% accuracy increase which achieves 49.13%. Table 1 and Table 2 illustrate the confusion matrix of the baseline system and our proposed system respectively. We can find that our proposed feature extraction method gets a significant improvement on the emotions ***Anger***, ***Neutral***, and ***Positive***. For emotion ***Emphatic***, we obtain a relative comparable performance. But for ***Rest***, the performance is somewhat low.

Table 1. Confusion Matrix of Baseline

		Rec				
		A	E	N	P	R
Ref	A	44.75%	15.50%	20.25%	10.75%	8.75%
	E	19.10%	34.10%	26.50%	6.00%	14.30%
	N	9.67%	14.96%	42.63%	15.33%	17.41%
	P	5.00%	6.00%	13.00%	59.00%	17.00%
	R	13.00%	13.67%	29.00%	19.67%	24.67%

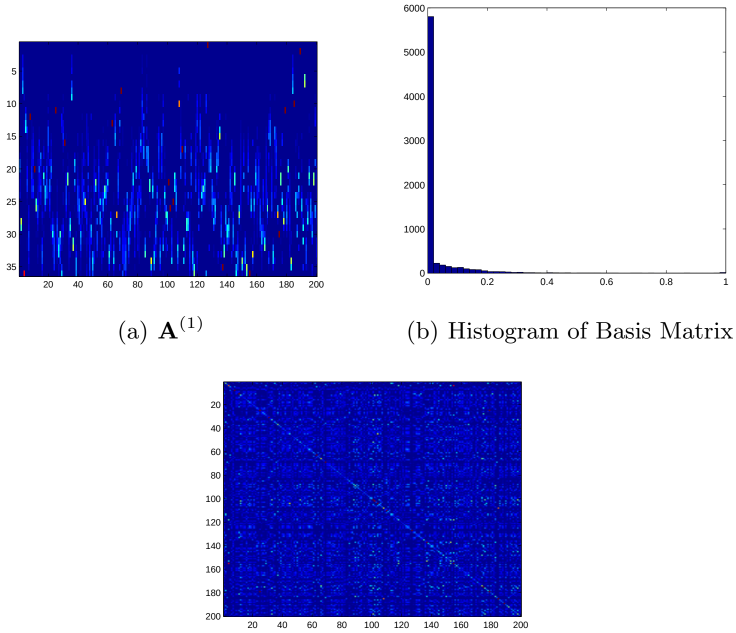
Table 2. Confusion Matrix of Proposed Method

		Rec				
		A	E	N	P	R
Ref	A	60.60%	9.48%	16.71%	3.24%	9.98%
	E	18.20%	30.10%	34.40%	3.80%	13.50%
	N	6.96%	11.22%	55.89%	17.22%	8.70%
	P	0.67%	0.00%	17.33%	69.33%	12.67%
	R	11.37%	9.03%	37.46%	26.09%	16.05%

Figure 4(a) gives us a view of the basis matrix $\mathbf{A}^{(1)}$. Figure 4(b) shows the histogram of the basis matrix. We can find that most values in the basis matrix are concentrated to zero or almost zero, and the sparse constraint takes a strong effect to the basis matrix. Figure 4(c) demonstrates the orthogonality of the basis matrix. From the result of $\mathbf{A}^{(1)T} \cdot \mathbf{A}^{(1)}$, we can observe that the diagonal values of the basis matrix are much larger than the others. It implies the balance between the orthogonality and reconstruction.

4 Conclusion

In this paper, we presents an experimental study of the feature extraction method based on constrained nonnegative tensor factorization using Gabor filtering for speech emotion recognition. Gabor filtering simulates the response of STRF in A1 of our auditory system and cNTF studies the basis matrices of the Gabor based auditory spectrum on tensor structure. Our proposed feature extraction method preserved more structural information underlying the speech signal by tensor representation and the sparse and orthogonal constraints remove more



(a) $\mathbf{A}^{(1)}$

(b) Histogram of Basis Matrix

(c) $\mathbf{A}^{(1)T} \cdot \mathbf{A}^{(1)}$

Fig. 4. A View of Basis Matrix $\mathbf{A}^{(1)}$

irrelevant information for classification. Experimental results show that our proposed feature extraction method gains a significant increased accuracy (49.13%) comparing with the MFCC based feature system (40.89%) on most of the emotion conditions.

Acknowledgment

The work was supported by the National Natural Science Foundation of China (Grant No. 60775007 and 90920014).

References

1. Vogt, T., André, E., Wagner, J.: Automatic recognition of emotions from speech: A review of the literature and recommendations for practical realisation. Affect and Emotion in Human-Computer Interaction: From Theory to Applications (2008) 75–91
2. Yacoub, S., Simske, S., Lin, X., Burns, J.: Recognition of emotions in interactive voice response systems. In: Eighth European conference on speech communication and technology. (2003)

3. Oudeyer, P.: The production and recognition of emotions in speech: features and algorithms. International Journal of Human-Computer Studies **59**(1-2) (2003) 157–183
4. Nwe, T., Foo, S., De Silva, L.: Speech emotion recognition using hidden Markov models. Speech communication **41**(4) (2003) 603–623
5. Neiberg, D., Elenius, K., Laskowski, K.: Emotion recognition in spontaneous speech using GMMs. In: Ninth International Conference on Spoken Language Processing. (2006)
6. Tato, R., Santos, R., Kompe, R., Pardo, J.: Emotional space improves emotion recognition. In: Seventh International Conference on Spoken Language Processing. (2002)
7. Sato, N., Obuchi, Y.: Emotion recognition using Mel-frequency cepstral coefficients. Information and Media Technologies **2**(3) (2007) 835–848
8. Bro, R.: PARAFAC. Tutorial and applications. Chemometrics and Intelligent Laboratory Systems **38**(2) (1997) 149–171
9. Kim, Y., Choi, S.: Nonnegative tucker decomposition. In: IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR'07. (2007) 1–8
10. Welling, M., Weber, M.: Positive tensor factorization. Pattern Recognition Letters **22**(12) (2001) 1255–1261
11. Qiu, A., Schreiner, C., Escabi, M.: Gabor analysis of auditory midbrain receptive fields: spectro-temporal and binaural composition. Journal of Neurophysiology **90**(1) (2003) 456
12. Chi, T., Ru, P., Shamma, S.: Multiresolution spectrotemporal analysis of complex sounds. The Journal of the Acoustical Society of America **118** (2005) 887
13. Wu, Q., Zhang, L., Shi, G.: Robust Speech Feature Extraction Based on Gabor Filtering and Tensor Factorization. In: ICASSP '09. (2009) 4649–4652
14. Wu, Q., Zhang, L., Shi, G.: Robust Feature Extraction for Speaker Recognition Based on Constrained Nonnegative Tensor Factorization. Journal of Computer Science and Technology **25(4)** (2010) 783–792
15. Pascual-Montano, A., Carazo, J., Kochi, K., Lehmann, D., Pascual-Marqui, R.: Nonsmooth nonnegative matrix factorization (nsNMF). IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(3) (2006) 403–415
16. Steidl, S.: Automatic Classification of Emotion-Related User States in Spontaneous Children's Speech. Logos Verlag, Berlin (2009)
17. Batliner, A., Steidl, S., Hacker, C., Nöth, E.: Private Emotions vs. Social Interaction – a Data-driven Approach towards Analysing Emotion in Speech. User Modeling and User-Adpated Interaction (umuai) **18, No. 1-2** (2008) 175–206

Online News Clustering System for Event Detection

Shun Wang[1] , Fang Li[1]

[1] Dept. Of Computer Science & Engineering
Shanghai Jiao Tong University, 800# Dong Chuan Rd.
Shanghai 200240, P.R. China.
fli@sjtu.edu.cn

Abstract. In the information age, there are huge amount of news stories on the Internet. People wish to learn what is going on and how it is going every day. This paper proposes an online news clustering system based on two-stage clustering algorithm. The first stage is micro-clustering for event detection, in which online news stories are clustered into micro-clusters. Then an event tracking process follows, where those new micro-clusters are compared with previous generated micro-clusters, either merged into old ones or be regarded as a new event. During the process, these micro-clusters are regarded either as event candidates or as an outlier (trivial events). The second stage is a macro-clustering algorithm, which runs on the result of micro-clustering to combine all candidates to its related events. Based on this two-stage clustering approach, the system can provide an overview of one specific event and its related events. The system has been realized and online for extracting every day's events from news stories.

Keywords: Event Detection, Clustering

1 Introduction

When people read some news stories on the Internet, they often have the needs to know the history or the future developments for some big events. However, this kind of requirement cannot be easily resolved by search engines with some simple queries [1]. Most of time, it is a frustrating experience for them. It would be helpful if news stories of one event could be grouped together. Therefore, some news websites manually compose reports for big events, such as: "The Tian An Warship Incident" and "The BP Oil Spill", which costs human labor and time to edit. There are still many important events as "International Incidents" or "Natural Disasters" that are not provided with all related stories.

The classic clustering algorithms fix the documents as a whole set, and try to process them once. It is not practical when processing online news stories, since news reports arrives as an endless documents stream.

Research of Topic Detection and Tracking (TDT) focused on newswires and broadcasts for many years. The state-of-the-art TDT techniques are still far from satisfying expectations [1]. In recent years, stream data clustering is investigated in Data Mining. It provides valuable algorithms and technologies to solve the problems.

In this paper, we use some ideas from stream data clustering. A two-stage clustering algorithm is proposed. The first stage is micro-clustering, in which online news reports are clustered into micro-clusters for event detection. Some micro-clusters are regarded as event candidates. Some are regarded as outliers which are trivial events or may be a seed story of future reports. Then we perform a macro-clustering for event tracking. Based on this two-stage clustering approach, our system can provide an overview of one specific event and all related events.

The following content is organized as follows: Section 2 gives a brief review of related work. Section 3 describes the details of the system. The experiments and result will be showed in Section 4 and a brief conclusion and discuss are in the final.

2 Related Work

Event detection and tracking are the aim of Topic Detection and Tracking (TDT) researches [2, 3, 4, 5, 6, 7]. There are still great efforts to make before in real use.

James Allan et.al has investigated and proposed some useful algorithms in his papers [3, 4, 5, 6]. Single pass clustering algorithm and its variants were widely used as a baseline in TDT benchmarks. Researchers in CMU used time window based single pass clustering algorithm for event detection [3]. In paper [2], the author used single pass clustering algorithm with two separated threshold TH_h and TH_l to determine whether a news stories is a new event or not. Dou Shen et.al compared different weight calculation and time window selection strategies when using single-pass algorithm in event detection [10]. Recently, new approaches for event detection using language model were proposed in paper [8, 9].

In the data stream clustering research, O'Callaghan L et.al proposed the STREAM algorithm to find clusters in the endless data stream [11, 12, 13]. Agrawal C C et.al proposed the CluStream algorithm to handle the defect of STREAM [14], which is applied to generate clusters for the evolving data stream. DenStream algorithm proposed by F. Cao et.al can find arbitrary shape clusters in the data stream with noises [15].

3 Event Detection with Two-Stage Clustering Algorithm

In our system, we follow the event definition in paper [16]. An event is defined as a set of stories that are similar to each other but different from stories in other event sets. The first step of our system is to collect news stories from various website. Then we perform preprocessing on the text. Each news report will be represented as a vector according to the dynamic TF-IDF model. Finally, News events will be generated based on two-stage clustering approach.

3.1 System Architecture

Figure 1 shows the system architecture of the proposed two-stage clustering. It consists of preprocessing, micro-clustering, event candidate selection and macro-clustering, which will be described in the following sections.

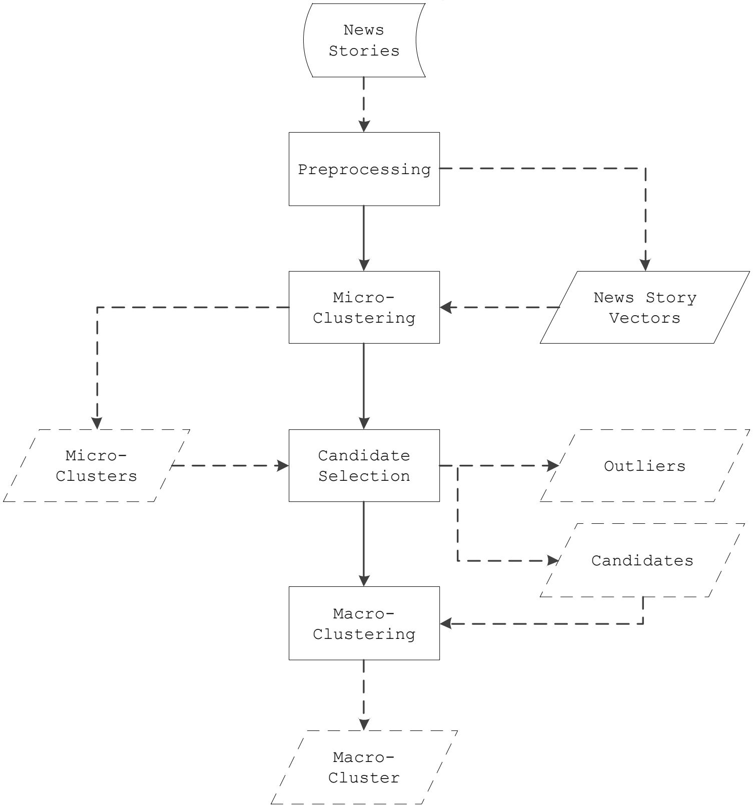


Fig. 1. System Architecture of Two-Stage Clustering

3.2 Preprocessing

HTML pages are fetched by crawlers and analyzed by parser to extract the title, contents as well as metadata such as publishing time, category, URL. Then, we perform word tokenize on stories' title and content. Part-of-speech tagging and named entities recognition are also done. Stop words are removed according to some part-of-speech tags, such as prepositions, conjunctions and so on.

3.3 Dynamic TF-IDF Model

In TDT tasks, Incremental TF-IDF model is widely used in term weight calculation [4, 7]. The DF (document frequency) of term *w* at time *t* is calculated as:

$$df_t(w) = df_{t-1}(w) + df_{c_t}(w) \; . \tag{1}$$

Where c_t means the collection of stories at time *t*, and $df_{ct}(w)$ means the document frequency of term *w* in c_t. $df_{t-1}(w)$ denotes the document frequency of term *w* in the time *t-1*. DF is updated incrementally at time *t*.

However, there are some drawbacks to process an infinite news stories. The weight of term *w* in time *t* is affected by the documents appearing before time *t* which may decrease the discrimination of keywords in the new stories. Since it is not possible to keep all documents in memory, the system will periodically remove old stories from memory. Therefore, the document frequency of term *w*, should reflect the current status of documents in the memory. For these reasons, we update DF dynamically. We proposed the dynamic TF-IDF model in formula 2.

$$df_t(w) = df_{t-1}(w) + df_{c_t}(w) - df_{c'_t}(w) \; . \tag{2}$$

Where $df_{c't}(w)$ represents the document frequency of term *w* in the corpus *c'* removed at time *t*. The DF of term w updates dynamically at time *t* with the buffer changes.

Then each story *d* coming at time *t* is represented as an n-dimension vector, where *n* is the number of distinct terms *w* in story *d*. Each dimension is weighted using dynamic TF-IDF model and the vector is normalized so that it is of unit length:

$$weight_t(d,w) = \frac{1}{Z_t(d)} tf(d,w) \cdot \log \frac{N_t}{df_t(w)} \; . \tag{3}$$

Where N_t is the total number of documents at time t. $Z_t(d)$ is a normalization value with:

$$Z_t(d) = \sqrt{\sum_w [tf(d,w) \cdot \log \frac{N_t}{df_t(w)}]^2} \; . \tag{4}$$

The similarity of two documents is calculated as:

$$sim_t^h(d,d') = \sqrt{weight_t(d,w) \cdot weight_t(d',w)} \; . \tag{5}$$

3.4 Micro-Clustering

At the beginning, new stories are clustered into new event candidates according to their pair-wise similarities. Every 500 news stories will be clustered online. Different from other clustering methods, we do micro-clustering algorithm on the new arrival stories rather than the old stories. We choose UPGMA (unweighted pair group method using arithmetic averages), an agglomerative clustering method, to perform micro-clustering process. The clustering procedure is described as below:

First, each new coming story is considered as a cluster. Then pair-wise distances between each two clusters are computed and stored in a distance matrix.

Second, suppose cluster C_1 and C_2 are two clusters with the minimum distance *d*. If *d* is less than the given threshold θ_d, The two clusters will be merged into a new cluster *C'*.

Third, adjust the new cluster's center vector to be the average of these two cluster's center vectors. Distance matrix is updated using unweighted arithmetic average method. If the minimum distance of two clusters is less than the given threshold θ_d," go back to the second step and continue. Otherwise the micro-clustering process ends.

3.5 Event Candidate Selection

After micro-clustering, a batch of micro-clusters is generated. Among them, there are lots of single story which cannot be grouped into any clusters. We consider these stories as a trivia event, i.e. outliers. Other clusters are regarded as event candidates.

Obvious, single story may be a seminal event, on which many later stories may follow. In this case, the system will find these seeds when relevant stories come into the system later. The candidate selection process consists of the following steps:

1) For each micro-cluster, get the most similar micro-cluster with the minimum pair-wise distance.

2) If the distance is less than the given threshold, merge these two micro-clusters into a new micro-cluster. This step will find the seminal story regarded as outlier in the micro-clustering. Then this new micro-cluster will be either put in the candidate buffer or outlier queue according to the number of documents it contains.

3) If the distance is greater than the given threshold θ_c, which means the new arrival micro-cluster is a truly new event candidate or an outlier. Then we directly put it either in the candidate buffer or in the outlier queue.

3.6 Macro-Clustering

After the candidate selection process, we get all the event candidates stored in our candidate buffer. In this step, we perform macro-clustering on all candidates in the candidate buffer to combine them into the news events. UPGMA clustering algorithm is also chosen for macro-clustering. Similarly, a designated threshold θ_m is used to control the clustering process. The final macro-clusters are supposed to be all the reports of one event.

4 Experiments and Results

4.1 Experimental Setup

In order to evaluate our system, we propose the following experiments:

1) Comparing with other clustering methods in order to describe why we choose the two-stage clustering algorithm.
2) Calculating the precision, recall and F-measure for macro-clustering and micro-clustering to show the effectiveness of the system.
3) Measuring the F-value for dynamic TF-IDF and incremental TF-IDF to show dynamic TF-IDF better than the incremental TF-IDF for large corpus.

We collect the corpus from the real web environment for the above experiments. The corpus contains 7479 news pages from the year 2007 to the year 2009. There are 50 events manually labeled during that period of time. These news reports are sorted in temporal and supposed incrementally arrived according to their time stamp during the experiments.

4.2 Evaluation Metric

We choose traditional evaluation metrics Precision, Recall which are widely used in Information Retrieval and Clustering. They are defined as:

$$recall(i,j) = \frac{n_{ij}}{n_i} . \tag{6}$$

$$precision(i,j) = \frac{n_{ij}}{n_j} . \tag{7}$$

Where n_{ij} is the number of stories of event i in cluster j, n_j is the number of stories in cluster j and n_i is the stories about event i.

The F-measure of cluster j and event i is given by[]:

$$F(i,j) = \frac{2 * recall(i,j) * precision(i,j)}{recall(i,j) + precision(i,j)} . \tag{8}$$

The F-measure of any event i is the maximum value of $F(i,j)$

$$F(i) = \max_{j=1,\ldots,m} (F(i,j)) . \tag{9}$$

Precision and Recall of event i , which is noted as *precision(i)* and *recall(i),* is the value which makes F-measure maximum.

An overall value for the *recall* and *precision* is calculated by taking the weighted average of all values for the corresponding metric as follows:

$$recall = \sum_i \frac{n_i}{n} recall(i) . \tag{10}$$

$$precision = \sum_i \frac{n_i}{n} precision(i) . \tag{11}$$

Where *n* is the total number of documents in all clusters.

4.3 Result and Discussion

The system incrementally processes news report from the corpus manually collected, by adding 500 news reports for each time. It needs 15 times to complete for 7479 news stories.

4.3.1 Comparing with Other Clustering Algorithms

We choose Single-Pass, Sphere K-means and DBSCAN for experimental comparisons. The Single-Pass and Sphere K-means are implemented by our lab. DBSCAN is from Weka. Since the Sphere K-means and DBSCAN clustering algorithm cannot run in incremental manner, they are run on different size of news reports for 15 times, the first time is on 500 reports, the second time is on 1000 reports, and the last time is on 7479 reports. The threshold of pair-wise distance for Single-Pass algorithm is set to 0.1. The K for Sphere K-means is set to 50 which is the number of total events in the corpus. The parameter of DBSCAN ε is set to 0.1, M is set to 5. For our two-stage clustering algorithm, the parameters in the following experiments are setup as: the θ_d is set to 0.1, θ_m is set to 0.3 and θ_c is set to 0.25. The size of candidate buffer is set to 300, and the outlier queue is set to 500. The result shows in the following figure, the x-axis is the number of news reports of each times, y-axis is the F-measure.

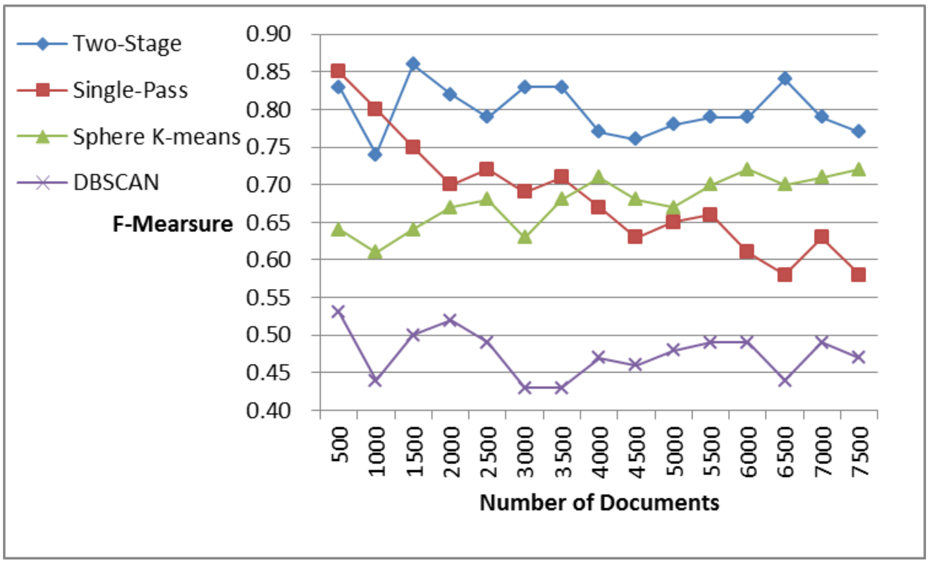


Fig. 2. F-Measure of Different Clustering Algorithm

Result shows that the F-Measure of Single-Pass is getting worse as the time with more and more news reports. The reason for single-pass is "error accumulation", which means mistake made in previous step affect the performance of the following steps. The Sphere K-means generate better F-Measure result with more news reports added to the system. The reason for this is that the K is fixed to 50 during 15 runs not considering the change of number of reports. DBSCAN performs worst in this experiment. This is caused by the data sparse in high dimension, which makes DBSCAN hard to find a density area. However, two-stage clustering can generate stabilized performance.

4.3.2 Evaluation of the two-stage clustering algorithm

Our two-stage clustering consists of micro-clustering and macro-clustering for event detection. The event tracking threshold and threshold of pair-wise distance for macro-clustering is set to 0.3. The precision, recall and F-measure of micro-clustering result of each time are shown in the Table 1:

Table 1. Evaluation of Micro-Clustering

No. of Times	No. of reports	No. of micro-clusters	No. of outliers	Precision	Recall	F-Measure
1	500	48	47	1.00	0.42	0.51
2	1000	87	92	1.00	0.52	0.64

3	1500	133	136	1.00	0.48	0.62
4	2000	189	185	0.99	0.49	0.63
5	2500	237	228	0.99	0.56	0.68
6	3000	281	271	0.99	0.56	0.68
7	3500	300	318	0.99	0.55	0.67
8	4000	300	360	0.99	0.61	0.72
9	4500	299	398	0.99	0.62	0.73
10	5000	300	433	0.98	0.60	0.71
11	5500	300	471	0.99	0.59	0.72
12	6000	300	500	0.99	0.42	0.55
13	6500	300	500	0.99	0.43	0.56
14	7000	296	500	0.98	0.40	0.54
15	7479	300	500	0.99	0.41	0.55

Table 1 shows that the system can generate high precision but low recall on micro-clusters. Micro-clustering are more favorable for precision, because merge step will be followed.

Figure 3 shows the performance of macro-clustering. With the increasing of documents, we can find out that our system can still generate satisfied F-Measure although the precision of our system gradually goes down.

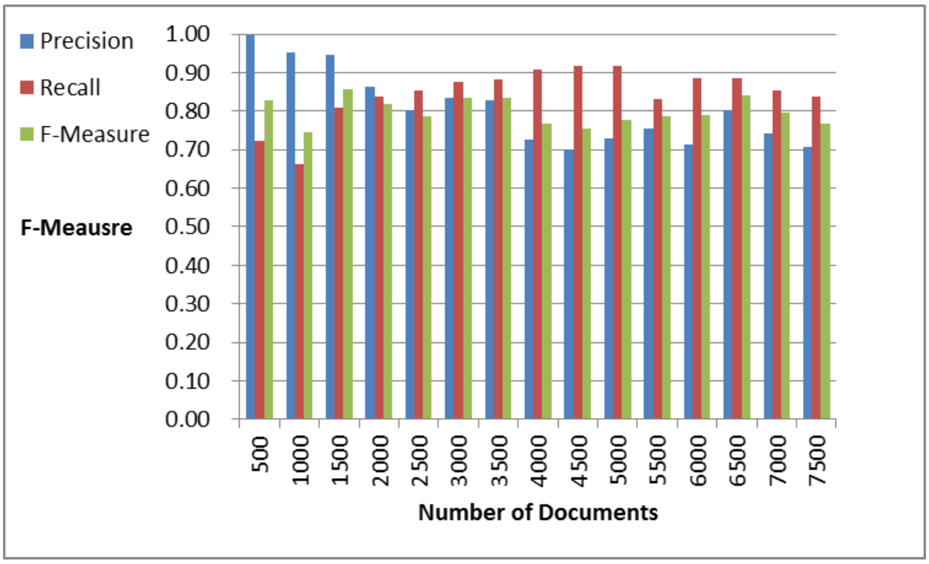


Fig. 3. Performance of Macro-Clustering

4.3.3 Comparison of Dynamic TF-IDF and incremental TF-IDF

The third experiment compares the performance of our Dynamic TF-IDF model with the classic TF-IDF model on test data. Figure 4 shows the result of F-measure. The experiment shows that when our system starts to delete some stories of inactive event, the dynamic TF-IDF model out performs the incremental TF-IDF model.

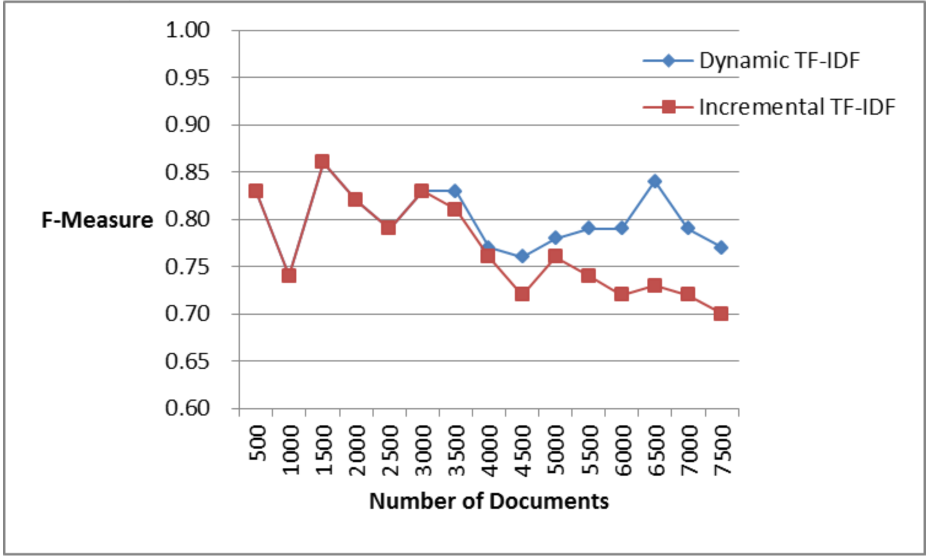
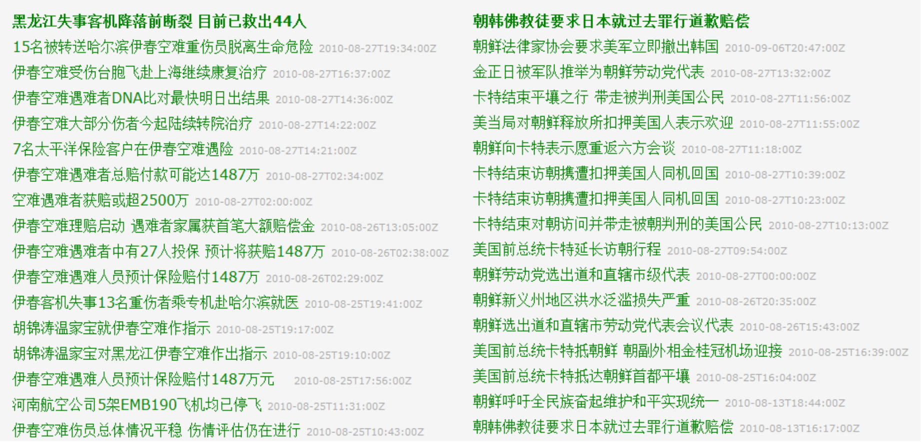


Fig. 4. F-Measure of Dynamic TF-IDF Model and Incremental TF-IDF Model

4.3.4 Discussion

Experiments show the effectiveness of our two-stage clustering algorithm. There are still some problems. Some result are topic clusters instead of event clusters, such as Figure 5a and Figure 5b. Figure 5a describes the air crash event of Yichun, while Figure 5b describes the topic of North-South Korean relations. To improve this problem, we will use more sophisticated algorithm to determine the pair-wise distance threshold for each macro-cluster rather than a universal threshold for all macro-clusters.



黑龙江失事客机降落前断裂 目前已救出44人
15名被转送哈尔滨伊春空难重伤员脱离生命危险 2010-08-27T19:34:00Z
伊春空难受伤台胞飞赴上海继续康复治疗 2010-08-27T16:37:00Z
伊春空难遇难者DNA比对最快明日出结果 2010-08-27T14:36:00Z
伊春空难大部分伤者今起陆续转院治疗 2010-08-27T14:22:00Z
7名太平洋保险客户在伊春空难遇险 2010-08-27T14:21:00Z
伊春空难遇难者总赔付款可能达1487万 2010-08-27T02:34:00Z
空难遇难者获赔或超2500万 2010-08-27T02:00:00Z
伊春空难理赔启动 遇难者家属获首笔大额赔偿金 2010-08-26T13:05:00Z
伊春空难遇难者中有27人投保 预计将获赔1487万 2010-08-26T02:38:00Z
伊春空难遇难人员预计保险赔付1487万 2010-08-26T02:29:00Z
伊春客机失事13名重伤者乘专机赴哈尔滨就医 2010-08-25T19:41:00Z
胡锦涛温家宝就伊春空难作指示 2010-08-25T19:17:00Z
胡锦涛温家宝对黑龙江伊春空难作出指示 2010-08-25T19:10:00Z
伊春空难遇难人员预计保险赔付1487万元 2010-08-25T17:56:00Z
河南航空公司5架EMB190飞机均已停飞 2010-08-25T11:31:00Z
伊春空难伤员总体情况平稳 伤情评估仍在进行 2010-08-25T10:43:00Z

Fig. 5a. Cluster for air crash in Yichun

朝韩佛教徒要求日本就过去罪行道歉赔偿
朝鲜法律家协会要求美军立即撤出韩国 2010-09-06T20:47:00Z
金正日被军队推举为朝鲜劳动党代表 2010-08-27T13:32:00Z
卡特结束平壤之行 带走被判刑美国公民 2010-08-27T11:56:00Z
美当局对朝鲜释放所扣押美国人表示欢迎 2010-08-27T11:55:00Z
朝鲜向卡特表示愿重返六方会谈 2010-08-27T11:18:00Z
卡特结束访朝携遭扣押美国人同机回国 2010-08-27T10:39:00Z
卡特结束访朝携遭扣押美国人同机回国 2010-08-27T10:23:00Z
卡特结束对朝访问并带走被朝判刑的美国公民 2010-08-27T10:13:00Z
美国前总统卡特延长访朝行程 2010-08-27T09:54:00Z
朝鲜劳动党选出道和直辖市级代表 2010-08-27T00:00:00Z
朝鲜新义州地区洪水泛滥损失严重 2010-08-26T20:35:00Z
朝鲜选出道和直辖市劳动党代表会议代表 2010-08-26T15:43:00Z
美国前总统卡特抵朝鲜 朝副外相金桂冠机场迎接 2010-08-25T16:39:00Z
美国前总统卡特抵达朝鲜首都平壤 2010-08-25T16:04:00Z
朝鲜呼吁全民族奋起维护和平实现统一 2010-08-13T18:44:00Z
朝韩佛教徒要求日本就过去罪行道歉赔偿 2010-08-13T16:17:00Z

Fig. 5b. Cluster for North-South Korean relations

5 Conclusions and Future Work

In this paper, an online news clustering algorithm for event detection is proposed. Based on the algorithm, an online news clustering system is implemented. It can be accessed via URL: http://lt-lab.sjtu.edu.cn:8989/CorpusSystem.

Our algorithm uses two-stage clustering methods. The first stage is micro-clustering for event detection, where online news reports are incrementally clustered into micro-clusters. It focuses on high precision. Then we perform an event tracking process, where those newly micro-clusters are compared with previous generated micro-clusters, either merged into old ones or be regarded as a new event. The second stage is a macro-clustering algorithm, to combine all candidates to its related events.

The proposed two-stage clustering algorithm enables our system processing infinite news stream in incremental manner. Meanwhile, with the help of high precision micro-clusters stored in our system as intermediate results, it is possible to rectify some mistakes in previous macro-clustering result in the next macro-clustering process.

In the next step we will focus on how to combine more sources, such as blogs and twitter, into our system, which facilitates our user to learn other people's attitude of one specific event.

6 References

[1] Canhui W., Min Z., Shaoping M., Liyun R.: Automatic online news issue construction in web environment. Proceeding of the 17th international conference on World Wide Web, 2008. 457~466

[2] Chen H., Lunwei K., Description of a topic detection algorithm on TDT3 mandarin text: proceedings of Topic Detection and Tracking Workshop. 2000. 165-166.

[3] James A.: Topic detection and tracking: event based information organization. Dordrecht: Kluwer Academic Publishers, 2002.

[4] James A., Ron P., Victor L.: On-line new event detection and tracking. In Proceeding of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Melbourne, Australia, 1998. 37~45

[5] James A, Ron P.: On-line New Event Detection, Clustering and Tracking [D]. Amherst: Department of Computer Science, UMASS,1999.

[6] James A, Jaime C., George D., Jonathan Y. et.al: Topic detection and tracking pilot study: Final report [A]. In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop [C], Virginia: Lansdowne, February 1998, 194-218.

[7] Yang Y., Pierce T., Carbonell J.: A study of retrospective and on-line event detection. In Proceeding of ACM SIGIR, Melbourne, August 24 28, 1998, pp.28-36.

[8] Xiaoyong L., W. Bruce C.: Cluster-based retrieval using language models. In Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Language models, 2004. 186-193

[9] Liu Y. B., Cai J.R., Yin J.: Clustering text data streams. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 23(1): 112-128 Jan. 2008

[10] Dou S., Qiang Y., Jiantao S., Zheng C.: Thread detection in dynamic text message streams. In Proc. ACM SIGIR 2006, Seattle, Washington, August 6-11, pp.35-42.

[11] O'Callaghan L., Mishra N., Meyerson A., Guha S.: Streaming data algorithms for high-quality clustering. Proceeding of ICDE 2002, San Jose, CA, February 26 March 1, pp.685-704.

[12] Guha S., Mishra N., Motwani R., O'Callaghan L.: Clustering data streams. In Proceeding of FOCS 2000, California, November 12 14, pp.359-366.

[13] Guha S., Meyerson A., Mishra N., Motwani R., O'Callaghan L.: Clustering data streams: Theory and practice. IEEE TKDE, 2003, 15(3):515-528.

[14] Charu C. A., Han J. W., Wang J., Yu P. S.: A framework for clustering evolving data streams. In Proc. VLDB 2003, Berlin, September 9 12, 2003, pp.81-92.

[15] Feng C., Martin E., Weining Q., Aoying. Z.: Density-Based Clustering over an Evolving Data Stream with Noise, SDM, 2006.

[16]Yang, C. C., Shi X.: Discovering event evolution graphs from newswires. *WWW*, 2006.