Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages, July 2003, pp. 138-145.

Text Categorization Using Automatically Acquired Domain Ontology

Shih-Hung Wu, Tzong-Han Tsai, Wen-Lian Hsu
Institute of Information Science
Academia Sinica
Nankang, Taipei, Taiwan, R.O.C.
shwu@iis.sinica.edu.tw, thtsai@iis.sinica.edu.tw, hsu@iis.sinica.edu.tw

Abstract

In this paper, we describe ontology-based text categorization in which the domain ontologies are automatically acquired through morphological rules and statistical methods. The ontology-based approach is a promising way for general information retrieval applications such as knowledge management or knowledge discovery. As a way to evaluate the quality of domain ontologies, we test our method through several experiments. Automatically acquired domain ontologies, with or without manual editing, have been used for text categorization. The results are quite satisfactory. Furthermore, we have developed an automatic method to evaluate the quality of our domain ontology.

1. Introduction

Domain ontology, consisting of important concepts and relationships of the concepts in the domain, is useful in a variety of applications (Gruber, 1993). However, evaluating the quality of domain ontologies is not straightforward. Reusing an ontology for several applications can be a practical method for evaluating domain ontology. Since text categorization is a general tool for information retrieval, knowledge management and knowledge discovery, we test the ability of domain ontology to categorize news clips in this paper.

Traditional IR methods use keyword distribution form a training corpus to assign testing document. However, using only keywords in a training set cannot guarantee satisfactory results since authors may use different keywords. We believe that, news clip events are categorized by concepts, not just keywords. Previous works shows that the latent semantic index (LSI) method and the n-gram method give good results for Chinese news categorization (Wu et al., 1998). However, the indices of LSI and n-grams are less meaningful semantically. The implicit rules acquired by these methods can be understood by computers, not humans. Thus, manual editing for exceptions and personalization are not possible and it is difficult to further reuse these indices for knowledge management.

With good domain ontology we can identify the concept structure of sentences in a document. Our idea is to compile the concepts within documents in a training set and use these concepts to understand documents in a testing set. However, building rigorous domain ontology is laborious and time-consuming. Previous works suggest that ontology acquisition is an iterative process, which includes keyword collection and structure reorganization. The ontology is revised, refined, and accumulated by a human editor at each iteration (Noy and McGuinness, 2001). For example, in order to find a hyponym of a keyword, the human editor must observe sentences containing this keyword and its related hyponyms (Hearst, 1992). The editor then deduces rules for finding more hyponyms of this keyword. At each iteration the editor refines the rules to obtain better quality pairs of keyword-hyponyms. To speed up

the above labor-intensive approach, semi-automatic approaches have been designed in which a human editor only has to verify the results of the acquisition (Maedche and Staab, 2000).

A knowledge representation framework, Information Map (InfoMap) in our previous work (Hsu et al., 2001), has been designed to integrate various linguistic, common-sense and domain knowledge. InfoMap is designed to perform natural language understanding, and applied to many application domains, such as question answering (QA), knowledge management and organization memory (Wu et al., 2002), and shows good results. An important characteristic of InfoMap is that it extracts events from a sentence by capturing the topic words, usually subject-verb pairs or hypernym-hyponym pairs, which are defined in the domain ontology.

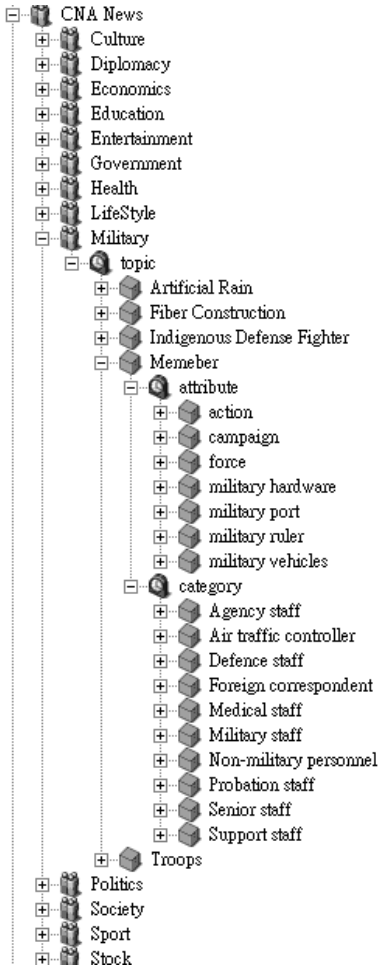

Figure 1. Ontology Structure for CNA News

We shall review the InfoMap ontology framework in Section 2. The ontology acquisition process and extraction rules will be introduced in Section 3. We describe ontology-based text categorization in Section 4. Experimental results are reported in Section 5. We conclude our work in Section 6.

2. Information Map

InfoMap can serve as domain ontology as well as an inference engine. InfoMap is designed for NLP applications; its basic function is to identify the event structure of a sentence. We shall briefly describe InfoMap in this section. Figure 1 gives example ontology of the Central News Agency (CNA), the target in our experiment.

2.1 InfoMap Structure Format

As a domain ontology, InfoMap consists of domain concepts and their related sub-concepts such as categories, attributes, activities. The relationships of a concept and its associated sub-concepts form a tree-like taxonomy. InfoMap also defines *references* to connect nodes from different branches which serves to integrate these hierarchical concepts into a network. InfoMap not only classifies concepts, but also connects the concepts by defining the relationships among them.

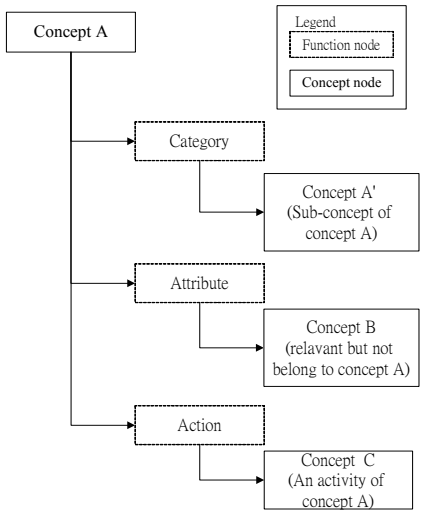

Figure 2. Skeleton of the Ontology Structure of InfoMap

In InfoMap, *concept nodes* represent concepts and *function nodes* represent the relationships between concepts. The root node of a domain is the name of the domain. Following the root node, important topics are stored in a hierarchical order. These topics have sub-categories that list related sub-topics in a recursive fashion. Figure 1 is a partial view of the domain ontology of the CNA. Under each domain there are several topics and each topic might have sub-concepts and associated attributes. In this example, note that, the domain ontology is automatically acquired from a domain corpus, hence the quality is poor. Figure 2 shows the skeleton order of a concept using InfoMap.

2.2 Event Structure

Since concepts that are semantically related are often clustered together, one can use InfoMap to discern the main event structure in a natural language sentence. The process of identifying the event structure, we call a firing mechanism, which matches words in a sentence to both concepts and relationships in InfoMap.

Suppose keywords of concept A and its sub-concept B (or its hyponyms) appear in a sentence. It is likely that the author is describing an event "B of A". For example, when the words "tire" and "car" appear in a sentence, normally this sentence would be about the tire of a car (not tire in the sense of fatigue). Therefore, a word-pair with a semantic relationship can give more concrete information than two words without a semantic relationship. Of course, certain syntactic constraints also need to be satisfied. This can be extended to a noun-verb pair or a combination of noun, verb and adjective. We call such words in a sentence an event structure. This mechanism seems to be especially effective for Chinese sentences.

2.3 Domain Speculation

With the help of domain ontologies, one can categorize a piece of text into a specific domain by categorizing each individual sentence within the text. There are many different ways to use domain ontology to categorize text. It can be used as a dictionary, as a keyword lists and as a structure to identify NL events. Take a single sentence for example. We first use InfoMap as a dictionary to do word segmentation (necessary for Chinese sentences) in which the ambiguity can be resolved by checking the domain topic in the ontology. After words are segmented, we can examine the distribution of these words in the ontology and effectively identify the densest cluster. Thus, we can use InfoMap to identify the domains of the sentences and their associated keywords. Section 4.1 will further elaborate on this.

3. Automatic Ontology Acquisition

The automatically domain ontology acquisition from a domain corpus has three steps:

1. Identify the domain keywords.
2. Find the relative concepts.
3. Merge the correlated activities.

3.1 Domain Keyword Identification

The first step of automatic domain ontology acquisition is to identify domain keywords. Identifying Chinese unknown words is difficult since the word boundary is not marked in Chinese corpus. According to an inspection of a 5 million word Chinese corpus (Chen et al., 1996), 3.51% of words are not listed in the CKIP lexicon (a Chinese lexicon with more than 80,000 entries). We use reoccurrence frequency and fan-out numbers to characterize words and their boundaries according to PAT-tree (Chien, 1999). We then adopt the TF/IDF classifier to choose domain keywords. The domain keywords serve as the seed topics in the ontology. We then apply SOAT to automatically obtain related concepts.

3.2 SOAT

To build the domain ontology for a new domain, we need to collect domain keywords and concepts by finding relationships among keywords. We adopt a semi-automatic domain ontology acquisition tool (SOAT, Wu et al., 2002), to construct a new ontology from a domain corpus. With a given domain corpus, SOAT can build a prototype of the domain ontology.

InfoMap uses two major relationships among concepts: taxonomic relationships (category and synonym) and non-taxonomic relationships (attribute and action). SOAT defines rules, which consist of patterns of keywords and variables, to capture these relationships. The extraction rules in

SOAT are morphological rules constructed from part-of-speech (POS) tagged phrase structure.

Here we briefly introduce the SOAT process:

Input: domain corpus with the POS tag
Output: domain ontology prototype
Steps:

1 Select a keyword (usually the name of the domain) in the corpus as the seed to form a potential root set *R*
2 Begin the following recursive process:
2.1 Pick a keyword *A* as the root from *R*
2.2 Find a new related keyword *B* of the root *A* by extraction rules and add it into the domain ontology according to the rules
2.3 If there is no more related keywords, remove *A* from *R*
2.4 Put *B* into the potential root set
Repeat step 2 until either *R* becomes empty or the total number of nodes reach a threshold

3.3 Morphological Rules

To find the relative words of a keyword, we check the context in the sentence from which the keyword appears. We can then find attributes or hyponyms of the keyword. For example, in a sentence, we find a noun in front of a keyword (say, *computer*) may form a specific kind of concept (say, *quantum computer*). A noun (say, *connector*) followed by "of" and a keyword may be an attribute of the keyword, (say, *connector of computer*). See (Wu et al., 2002) for details.

3.4 Ontology Merging

Ontologies can be created by merging different resources. One NLP resource that we will merge into our domain ontology is the noun-verb event frame (NVEF) database (Tsai and Hsu, 2002). NVEF is a collection of permissible noun-verb sense-pairs that appear in general domain corpora. The noun will be the subject or object of the verb. This noun-verb sense-pair collection is domain independent. We can use nouns as domain keywords and find their correlated verbs. Adding these verbs into the domain ontology makes the ontology more suitable for NLP. The correlated verbs are added under the *action* function node.

4. Ontology-Based Text Categorization

To incorporate the domain ontology into a text categorization, we have to adjust both the training process and testing process. Section 4.1 describes how to make use of the ontology and the event structure during the training process. Section 4.2 describes how to use ontology to perform domain speculation. Section 4.3 describes how to categorize news clippings.

4.1 Feature and Threshold Selection

With the event structure matched (fired) in the domain ontology, we have more features with which to index a text. To select useful features and a proper threshold, we apply Microsoft Decision Tree Algorithm to determine a path's relevance as this algorithm can extract human interpretable rules (Soni et al., 2000).

Features of the event structure include event structure score, node score, fired node level, and node type. During the training process, we record all features of the event structure fired by the news clippings in the domain-categorized training corpus. The decision tree shows that a threshold of 0.85 is sufficient to evaluate event structure scores. We use event structure score to determine if the path is relevant. According to Figure 3, if the threshold of true probability is 85%, then the event structure score (*Pathscore* in the figure) should be 65.75. And the relevance of a path p is true if p falls in a node on the decision tree whose ratio of true instance is greater than λ.

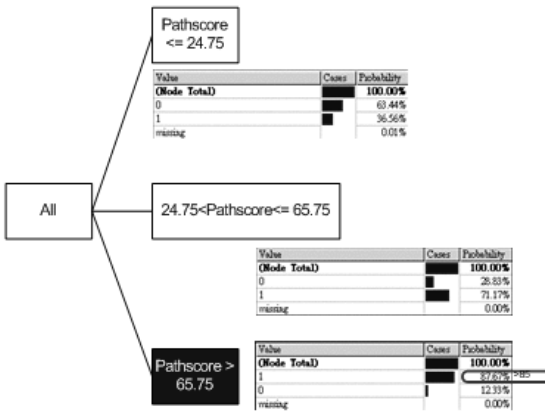

Figure 3. Threshold selection using decision tree

4.2 Domain Speculation

The goal of domain speculation is to categorize a sentence S into a domain D_j according to the combined score of the keywords and the event structure in sentence S. We first calculate the similarity score of S and D_j. The keyword score and the event structure score are calculated independently.

$$SimScore(D_j, S) = Keyword\_Score(D_j, S) + \alpha * EventStructure\_Score(D_j, S)$$

We use the TF/IDF classifier (Salton, 1989) to calculate the *Keyword_Score* of a sentence as follows. First, we use a segmentation module to split a Chinese sentence into words. The TF/IDF classifier represents a domain as a weighted vector, $D_j = (w_{j1}, w_{j2}, \ldots, w_{jn})$, where n is the number of words in this domain and w_k is the weight of word k. w_k is defined as $nf_{jk} * idf_{jk}$, where nf_{jk} is the term frequency (i.e., the number of times the word w_k occurs in the domain j). Let DF_k be the number of domains in which word k appears and $|D|$ the total number of domains. idf_k, the inverse document frequency, is given by:

$$idf_k = \log(\frac{|D|}{DF_k}).$$

This weighting function assigns high values to domain-specific words, i.e. words which appear frequently in one domain and infrequently in others. Conversely, it will assign low weights to words appearing in many domains. The similarity between a domain j and a sentence represented by a vector D_i is measured by the following cosine:

$$Keyword\_Score(D_j, S) = Sim(D_j, D_i) = \frac{\sum_{k=1}^{n} w_{jk} w_{ik}}{\sqrt{\sum_{k=1}^{n} (w_{jk})^2 \sum_{k=1}^{n} (w_{ik})^2}}$$

The event structure score is calculated by InfoMap Engine. First, find all the nodes in ontology that match the words in the sentence. Then determine if there is any concept-attribute pair, or hypernym-hyponym pair. Finally, assign a score to each fired event structure according to the string length of words that match the nodes in the ontology. The selected event structure is the one with the highest score.

$$EventStructure\_Score(D_j, S) = \max_{Event} \sum StringLength(keywords(D_j \cap S))$$

4.3 News Categorization

Upon receiving a news clipping C, we split it into sentences S_i. The sentences are scored and categorized according to domains. Thus, every sentence has an individual score for each domain $Score(D, S_i)$. We add up these scores of every sentence in the text according to domain, giving us total domain scores for the entire text. The domain which has the highest score is the domain into which the text is categorized.

$$Domain(C) = \arg\max_{D} (\sum_{S \in C} Score(D, S_i))$$

5. Refining Ontology through the Text Categorization Application

The advantage of ontology compared to other implicit knowledge representation mechanism is that it can be read, interpreted and edited by human. Noise and errors can be detected and refined, especially for the automatically acquired ontology, in order to obtain a better ontology. Another advantage of allowing human editing is that the ontology produced can be shared by various applications, such as from a QA system to a knowledge management system. In contrast, the implicit knowledge represented in LSI or other representations is difficult to port from one application to another.

In this section, we show how the human editing feature improves news categorization. First, we can identify a common error type: ambiguity; then, depending on the degree of categorization ambiguity, the system can report to a human editor the possible errors of certain concepts in the domain ontology as clues.

Consider the following common error type: event structure ambiguity. Some event structures are located in several domains due to the noise of training data. We define two formulas to find such event structures. The ambiguity of an event structure $E(S_i)$ is proportional to the number of domains in which it appears, and inversely proportional to its event score, where S_i are the sentences that fire event E.

$$GlobalCategorizationAmiguityFactor(E(S_i)) = number\ of\ domains\ fired\ by\ S_i/average(EventScore(S_i))$$

We also measure the similarity between every two event structures by calculating the co-occurrence multiplied by the global categorization ambiguity factor.

$$GlobalCategorizationAmbiguity_{ij}(E_i, E_j) = \text{Co-occurrence}(E_i, E_j) * GlobalCategorizationAmbiguityFactor(E_j)$$

When the GlobalCategorizationAmbiguity of an event structure E_i exceeds a threshold, the system will suggest that the human editor refine the ontology.

6. Experiments

To assess the power of domain identification of ontology, we test the text categorization ability on two different corpora. The ontology of the first experiment is edited manually; the ontology of the second experiment is automatically acquired. And we also conduct an experiment on the effect of human editing of the automatically acquired ontology.

6.1 Single Sentence Test

We test 9,143 sentences, edited manually for a QA system. The accuracy is 94%. These sentences are questions in the financial domain. Because the sentence topics are quite focused, the accuracy is very high. See Table 1.

Table 1. Sentence Categorization Accuracy

Domain #	Sentence #	Accuracy
24	9143	94.01%

6.2 News Clippings Collection

The second experiment that we conduct is news categorization. We collect daily news from China News Agency (CNA) ranging from 1991 to 1999. Each news clipping is short with 352 Chinese characters (about 150 words) on the average. There are more than thirty domains and we choose 10 major categories for the experiment.

6.3 10 Categories News Categorization

Our ten categories are: domestic arts and education (DD), foreign affairs (FA), finance report (FX), domestic health (HD), Taiwan local news (LD), Taiwan sports (LD), domestic military (MD), domestic politics (PD), Taiwan stock markets (SD), and weather report (WE). From each category, we choose the first 100 news clippings as the training set and the following 100 news clippings as the testing set. After data cleansing, the total training set has 979 news clippings, with 27,951 nodes and less than 10,000 distinct words. The training set for which domain ontologies are automatically acquired is shown in Table 2. A partial view of this ontology is in Figure 1.

The result of text categorization based on this automatically acquired domain ontology is shown in Table 5, which contains the recall and precision for each domain. Note that, without the help of the event structure, the macro average f-score is 85.16%. Even the total number of domain key concepts is less than 10,000 words (instead of 100,000 words in standard dictionary), we can still obtain a good categorization result. With the help of event structure, the macro average f-score is 85.55%.

6.4 Human Editing

To verify the refinement method, we conduct an experiment to compare the result of using automatically acquired domain ontology and that of limited human editing (on only one domain ontology). After the training process, we use domain ontologies to classify the training data, and to calculate the global categorization ambiguity factor formula in order to obtain

ambiguous event structure pairs as candidates for human editing. For simplicity, we restrict the action of refinement to *deletion*. It takes a human editor one half day to finish the task and delete 0.62% nodes (172 out of 27,951 nodes). In the testing phase, we select 928 new news clippings as the testing set. Table 3 shows the results from before and after human editing. Due to time constraints, we only edit the part of the ontology that might affect domain DD. The recall and precision of domain DD increase as well as both the average recall and average precision. In addition, the recall of domains having higher correlation with DD, such as PD and FA, decreases. Apparently, the event structures that mislead the categorization system to theses domain have mostly been deleted. The experiment result is very consistent with our intuition.

Table 2. Ten Category training set CNA news

Domain	Training set size	
	Doc#	Char#
DD	98	41870
FA	97	38143
FX	100	30771
HD	96	39818
JD	107	35381
LD	96	36957
MD	89	32903
PD	100	43152
SD	109	33030
WE	87	30457
total	979	362,482

7. Discussions and Conclusions

Compared to an ordinary n-gram dictionary, our ontology dictionary is quite small (roughly 10%) but records certain important relations between keywords.

Our goal is to generate rules that are human readable via ontology. The experiment result shows that event structure enhances text categorization, even when the domain ontology is automatically acquired without human verification. To improve our ontological approach, our future work are: 1. human editing in more domains; 2. enlarge our dictionary by merging existing ontologies, e.g., the names of countries, capitals and important persons, which are absent from the training corpus; 3. incorporate more sense pairs such as N-A (noun-adjective), Adv-V (adverb-verb); 4. use machine learning model on the weighting of the ontological features.

Previous research shows that some NLP techniques can improve information retrieval. Ontology-based IR is one of them. However, the construction of domain ontology is too costly. Thus, automatic acquisition of domain ontology is becoming an interesting research topic. Previous research shows that implicit rules (such as LSI, N-gram dictionaries) learned from a training corpus give better results than explicit rules generated by humans. However, it is hard to use these implicit rules or to combine them with other resources for further refinement. With the help of domain ontology, we can automatically generate rules that humans can understand. Since humans and machines can maintain ontology independently, the ontological approach can be applied more easily to other IR applications. Ontologies from different sources can be merged into the domain ontology. The system should include an editing interface that human thoughts can be incorporated to complement statistical rules. With semi-automatically acquired domain ontology, text categorization can be adapted to personal preferences.

8. References

Chen, K.J., C.R. Huang, L.P. Chang & H.L. Hsu, SINICA CORPUS: Design Methodology for Balanced Corpora, in Proceedings of PACLIC 11th Conference, pp.167-176, 1996.

Chien, L.F., PAT-tree-based Adaptive keyphrase extraction for Intelligent Chinese Information Retrieval, Information Processing and Management, Vol. 35, pp. 501-521, 1999.

Gruber, T.R. (1993), A translation approach to portable ontologies. Knowledge Acquisition, 5(2), pp. 199-220, 1993.

Hearst, M.A. (1992), Automatic acquisition of hyponyms from large text corpora. In COLING-92, pp. 539-545.

Hsu, W.L., Wu, S.H. and Chen, Y.S., Event Identification Based On The Information Map - INFOMAP, in Natural Language Processing and Knowledge Engineering Symposium of the IEEE Systems, Man, and Cybernetics Conference, Tucson, Arizona, USA, 2001.

Maedche, A. and Staab, S. (2000), Discovering Conceptual Relationships from Text. In: Horn,

W. (ed.): ECAI 2000. Proceedings of the 14th European Conference on Artificial Intelligence, IOS Press, Amsterdam.

Noy, N.F. and McGuinness D.L. (2001), Ontology Development 101: A Guide to Creating Your First Ontology, SMI technical report SMI-2001-0880, Stanford Medical Informatics.

Salton, G., Automatic Text Processing, Addison-Wesley, Massachusetts, 1989.

Soni, S, Tang, Z. and Yang, J., "Microsoft Performance Study of Microsoft Data Mining Algorithms", UniSys, 2000/12.

Tsai, J.L. and Hsu, W.L., "Applying an NVEF Word-Pair Identifier to the Chinese Syllable-to-Word Conversion Problem," COLING-02, Taipei, ACM press, 2002.

Wu, S.H. and Hsu, W.L., SOAT: A Semi-Automatic Domain Ontology Acquisition Tool from Chinese Corpus, COLING-02, Taipei, ACM press, 2002.

Wu, S.H., Day, M.Y., Tsai, T.H. and Hsu, W.L., FAQ-centered Organizational Memory, in Nada Matta and Rose Dieng-Kuntz (ed.), Knowledge Management and Organizational Memories, Kluwer Academic Publishers, Boston, 2002.

Wu, S.H., Yang, P.C. and Soo, V.W., An Assessment on Character-based Chinese News Filtering Using Latent Semantic Indexing, Computational Linguistics & Chinese Language Processing, Vol. 3, no.2, August 1998.

Table 3. Experiment result of CNA news categorization

Domain	# of nodes automatically acquired		#of nodes deleted in human editing		TF/IDF(baseline)			TF/IDF+Event Structure(first improvement)			TF/IDF+Event Structure with Human Editing (second improvement)			The different between (second improvement) and (first improvement)		
	Before	After	#	%	P%	R%	F%	P%	R%	F%	P%	R%	F%	P+%	R+%	F+%
DD	4616	4574	42	0.91	72.90	82.98	77.61	74.04	81.91	77.78	74.29	82.98	78.39	0.25	1.07	0.61
FA	8352	8348	4	0.05	75.83	94.79	84.26	71.32	95.83	81.78	76.67	95.83	85.19	5.35	0.00	3.41
FX	44	44	0	0.00	100	100	100	100	100	100	100	100	100	0.00	0.00	0.00
HD	3357	3348	9	0.27	78.79	88.64	83.42	80.21	87.50	83.70	78.79	88.64	83.42	-1.42	1.14	-0.28
JD	1854	1846	8	0.43	88	71.74	79.04	87.18	73.91	80	87.84	70.65	78.31	0.66	-3.26	-1.69
LD	2925	2831	94	3.21	87.64	80.41	83.87	90.36	77.32	83.33	88.51	79.38	83.70	-1.85	2.06	0.37
MD	2010	1999	11	0.55	95.59	66.33	78.31	95.71	68.37	79.76	97.26	72.45	83.04	1.55	4.08	3.28
PD	3199	3195	4	0.13	65.81	68.75	67.25	70.43	72.32	71.37	66.67	69.64	68.12	-3.76	-2.68	-3.25
SD	585	585	0	0.00	100	100	100	100	100	100	100	100	100	0.00	0.00	0.00
WE	1009	1009	0	0.00	95.74	100	97.83	95.74	100	97.83	95.74	100	97.83	0.00	0.00	0.00
Total	27951	27779	172	0.62												
Macro Average					86.03	85.36	85.16	86.50	85.72	**85.55**	86.58	85.96	**85.80**	0.08	0.24	**0.25**