Context-Based Retrieval System for Similar Medical Documents

Kazuya OKAMOTO
Graduate School of Informatics, Kyoto University, Kyoto, Japan.
kazuya@kuhp.kyoto-u.ac.jp

Tadamasa TAKEMURA
Department of Medical Informatics, Kyoto University Hospital, Kyoto, Japan.
takemura@kuhp.kyoto-u.ac.jp

Tomohiro KURODA
Department of Medical Informatics, Kyoto University Hospital, Kyoto, Japan.
Tomohiro.Kuroda@kuhp.kyoto-u.ac.jp

Hiroyuki YOSHIHARA
Department of Medical Informatics, Kyoto University Hospital, Kyoto, Japan.
lob@kuhp.kyoto-u.ac.jp

Abstract

This paper proposed a context-based similarity measurement model for retrieving discharge summaries and tried to retrieve similar discharge summaries. A discharge summary consists of sentences that have an attribute, one of clinical cycle consisting of "observation", "diagnosis" and "therapy". The model understands a discharge summary as continuation of "observation", "diagnosis" and "therapy", and measures similarities based on the context between the documents. This paper evaluated the model in two retrieval processes, a matching process and a ranking process. There was not so much difference in a matching process between the model and a vector space model that measures similarities by a number of kinds of words included in two compared documents. In contrast, the proposed model was superior to the vector space model in a ranking process. This paper concluded that the context-based model should be adapted with the vector space model executed fast for effective retrieval system.

1. Introduction

The use of electronic patients' record systems has been realized in many hospitals. Such systems make it easier to retrieve information than referencing paper records and medical documents have been pooling.

With the pooling of medical documents, Natural Language Processing has been getting important to analyze the pooled documents in order to produce fruitful knowledge and findings.

One of fruitful adaptations of Natural Language Processing to medical documents is document retrieval system to find out patient records of similar phenomena to a patient under treatment. Clinicians can refer previous patients with similar syndromes by giving a patient record of the patient under treatment as a query for the system. Browsing similar records, doctors may confirm his medical process to go through, and residents can find overlooked issues.

However similar document retrieval system does not work properly in general. In informatics, relevance feedback was tried to improving retrieval performance [1] and utilizing metadata or annotations as human knowledge has been tried to make an improvement [2]. In contrast, full use of characteristics of medical documents may result in efficient similar document retrieval system for medical documents.

In this article, we exploit characteristics of medical documents for efficient similar document retrieval system. The characteristics are as follows.

- Medical documents mostly consist of medical descriptions.
- Medical documents have clinical cycles.

The first feature gives certain limitation of vocabulary to explain a certain phenomena. Thus, introduction of medical term dictionaries should make similarity analysis easier on the contrast of general documents. So we introduce a medical term dictionary that we developed. Additionally, the first feature gives medical documents a certain manner of writing reflecting clinical process itself. As shown in second feature, clinical process is performed in cycle-wise, the document itself has a certain resemblance to basic clinical cycle. So, we introduce analysis to find out "clinical cycle" from given medical documents to make resemblance more clearly. The clinical cycle is comprised of three attributes: "observation", "diagnosis" and "therapy" [3]. A doctor makes observation of his patient and he uses the observation to make diagnosis. He provides therapy according to his diagnosis. And his therapy leads to next observation. Therefore, similarities of documents are considered related to context of medical documents. We propose a context-based similarity measurement model for retrieving similar medical documents. The proposed model classifies

segments of medical documents into observation, diagnosis and therapy exploiting a statistical learning technique, a SVM (a Support Vector Machine), and a statistical measure, TF (Term Frequency) / IDF (Inverse Document Frequency). And the proposed model finds context of each document based on the classifications and measures context-based similarities between documents.

The proposed model is evaluated in two experiments because similarities between the documents are referred to in a matching retrieval process and a ranking retrieval process. The experiments compare the proposed model and a vector space model that is used generally to measure similarities or is exploited to classify documents [4]. The results validate effectiveness of similar medical document retrieving system that exploits the proposed model.

2. Materials

2.1. Retrieval medical documents

Retrieval medical documents are medical history data and clinical process data of discharge summaries described by MML (Medical Markup Language) [5], which is an XML-based language and a medical information exchange format. Discharge summaries are for doctors to share patient medical records with other doctors or co-medical staff. Discharge summaries must be summarized frankly and briefly to share essential knowledge. Because of this characteristic, discharge summaries may be regarded as knowledge source. Figure 1 shows an example of a discharge summary.

In this article, retrieval medical documents are composed of <mmlSm:history> items and <mmlSm:clinicalCourse> items, which are medical history data and clinical process data respectively.

Medical documents are not always described by MML. For example, HL7 [6] is one of the other representative XML-based languages to describe medical records. But diverse data such as medical history data and clinical process data should be described by natural language as elements of each XML-based language because of its diversity. Therefore, the methods in this article do not depend on MML.

2.2. Japanese morphologic analysis

Word boundaries are not clear in non-segmented languages such as Japanese and Chinese. In such languages, the morphological analyzer of the corresponding non-segmented language must identify word segmentation.

To identify word segmentation in Japanese, a Japanese morphological analyzer needs dictionaries for guidewords and parts of speech. Furthermore, since medical documents have a lot of medical terms, the analyzer must use medical dictionaries to deal with medical documents. We add medical dictionaries [7] that have extracted medical terms from some kinds of medical documents such as the Merck Manuals [8] to ChaSen [9], the major Japanese morphological analyzer.

ChaSen exploits dictionaries for guidewords and parts of speech and Japanese syntax. For example, nouns can follow adjectives. In actuality, it is difficult to identify word segmentation because of complexity of Japanese.

```
  </mmlSm:patientProfile>
- <mmlSm:history>
    20代より動悸を自覚し、不整脈を指摘されたが加療は行っていなかった。
    <xhtml:br />
    2002年8月3日右片麻痺、左半身の痛覚低下、嚥下障害を呈した脳梗塞(延髄外側症候群)で入院。入院時心電図にて心房粗動を認め自然に洞調律へ回復した。また心原性塞栓の鑑別のため経食道心エコーがなされ卵円孔開存を指摘された。歩行器歩行可能となり同年8月23日リハビリのため転院した。以後後遺症は全快し自宅退院。
    <xhtml:br />
    ...
  </mmlSm:history>
- <mmlSm:physicalExam>
    【身体所見】身長178.0cm,体重,BMI,体温37.2℃,脈拍数100/分,呼吸数16/分,血圧(右)132/81(左)/mmHg
    <xhtml:br />
    【輸血情報】輸血なし
    <xhtml:br />
  </mmlSm:physicalExam>
- <mmlSm:clinicalCourse>
    <mmlSm:clinicalRecord> 4月7日入院後心電図モニタリングを行い経過観察。動悸やモニター上頻拍発作なく経過。血液検査所見も異常なく4月8日Holter心電図評価でもATなど頻脈発作の記録は無かった。当院不整脈外来でEPS, ablationについて検討予定とし4月9日退院となった。
    </mmlSm:clinicalRecord>
  </mmlSm:clinicalCourse>
  <mmlSm:dischargeFindings>シベノール300mg3×にて洞調
```

Figure 1. An example of a discharge summary

2.3. A SVM

A SVM is a statistical learning technique. Boser, Guyon and Vapnik proposed a SVM [10]. A SVM finds an optimal hyperplane separating training samples each of which is in vector space and has a positive or negative class.

Let us define training data that can be separated (Figure 2). Training samples is $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_s, y_s)$. $\mathbf{x}_i$ is a feature vector of the i-th sample and an N dimensional vector. y_i is the class label of the i-th sample and positive (+1) or negative (-1). s is the number of the training samples. A hyperplane is shown as follows.

$$\langle \mathbf{w} \cdot \mathbf{x} \rangle + b = 0 \qquad (1)$$

$\mathbf{w}$ is an N dimensional vector and b is a variable. Under the conditions, a SVM solves the following problem.

$$\begin{aligned} &\text{minimise} \quad \langle \mathbf{w} \cdot \mathbf{w} \rangle \\ &\text{subject to} \quad yi\left(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b\right) \geq 1, \quad i = 1, \ldots, s. \end{aligned} \qquad (2)$$

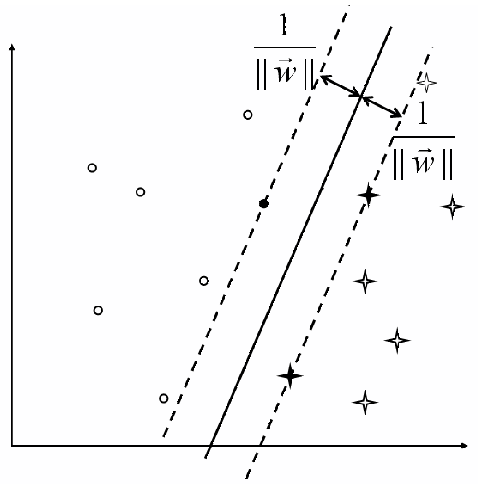

Figure 2. Training data that can be separated

In practice, a SVM is not able to build the separating hyperplane when training samples cannot be separated linearly because of some noise data. In this case, a SVM relaxes the above conditions introducing variable ξ_i as follows.

$$\text{minimise} \quad \langle \mathbf{w} \cdot \mathbf{w} \rangle + C \sum_{i=1}^{s} \xi_i$$

$$\text{subject to} \quad yi\left(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b\right) \geq 1 - \xi_i, \quad i = 1, \ldots, s. \quad (3)$$

C is a constant.

In this article, we use Tiny-SVM [11], one of the most available SVMs.

2.4. TF/IDF

TF and IDF are measures of documents' weights to retrieve documents with keywords. TF shows how many keywords are in a document. Behind this weighting is the fact that Luhn has described "a repeated concept is important" [12]. However, a word that is a generally high-frequency word is not important to characterize a document [13] and keywords' weights have been needed.

IDF shows a keyword's weight that is calculated as follows.

$$idf(t) = \ln \frac{N}{df(t)} \quad (4)$$

idf(*t*) shows a weight of keyword *t*. N shows a number of retrieval documents. *df*(*t*) shows a number of documents that include keyword *t*. *idf*(*t*) is largest if only one document includes keyword *t*, and *idf*(*t*) is smallest if all documents include keyword *t*. TF/IDF is a measure that shows total value of IDFs of keywords included in a document.

3. Methods

The proposed model exploits a SVM and TF/IDF to label clauses of medical documents as observation, diagnosis and therapy. Exploiting the labels, the model can find clauses that have strong anteroposterior relations. Based on the relations, the model measures the similarities between the medical documents.

3.1. ODT cycles

The proposed model classifies clauses of medical documents into observation, diagnosis and therapy statistically. Observation, diagnosis and therapy are abbreviated as O, D and T respectively. Each of O, D and T is called an ODT element and a cycle of O, D and T is called an ODT cycle. We count one cycle if each of O, D and T is included only once. Figure 3 shows ODT cycles.

When he was admitted to the hospital, all drugs including rimatil were withdrawn. DOE exacerbation, exacerbation X-ray shadow of his chest are associated with a time when he took rimatil and I doubted his drug associated hepatitis. His DLST of rimatil, and his patch test of rimatil were negative.

入院時より、すでにリマチルを含むすべての薬剤はOFFになっていた。DOEの増悪、胸部レントゲンの陰影の増悪がリマチル内服時に一致していたことよりリマチルによる薬剤性肺炎を疑った。リマチルのDLST, パッチテストでは陰性であった。…

↓

T: When he was admitted to the hospital, all drugs including rimatil were withdrawn.
O: DOE exacerbation, exacerbation of X-ray shadow of his chest are associated with a time when he took rimatil
D: and I doubted his drug associated hepatitis.
O: His DLST of rimatil, and his patch test of rimatil were negative.
…

Figure 3. ODT cycles

3.1.1. Training sets. We make two kinds of training sets. A SVM should learn one kind of training sets. The training set is medical documents that are segmented by Japanese pause marks into clauses labeled as O, D and T correctly by personal and one clause may have two or three labels of O, D and T. Figure 4 shows an example of the training set.

When he was admitted to the hospital, all drugs including rimatil were withdrawn. DOE exacerbation, exacerbation X-ray shadow of his chest are associated with a time when he took rimatil and I doubted his drug associated hepatitis. His DLST of rimatil, and his patch test of rimatil were negative.

入院時より、すでにリマチルを含むすべての薬剤はOFFになっていた。DOEの増悪、胸部レントゲンの陰影の増悪がリマチル内服時に一致していたことよりリマチルによる薬剤性肺炎を疑った。リマチルのDLST, パッチテストでは陰性であった。…

↓

T: When he was admitted to the hospital,
T: all drugs including rimatil were withdrawn.
O: DOE exacerbation,
OD: exacerbation of X-ray shadow of his chest are associated with a time when he took rimatil and I doubted his drug associated hepatitis.
O: His DLST of rimatil,
O: and his patch test of rimatil were negative.
…

Figure 4. An example of the training set that a SVM should learn

The other kind of training sets leads to TF/IDF. The training set is medical documents that are segmented by

personal into clauses and each clause is labeled by personal to have only one or no label. Figure 3 is also an example of the training set.

3.1.2. Automatic classification utilizing a SVM. A SVM learns a training set that is segmented by Japanese pause marks into clauses each of which may have two or three labels of O, D and T. First, ChaSen, Japanese morphologic analysis, divides the training set into words. And the proposed model puts each clause into vector space each axis of which corresponds one noun or one verb, and vector values are numbers of the corresponding words. If a clause is "He caught a cold", a value of axis "cold" is one. If a clause is "administering medicine A 50mg and medicine B 100mg", a value of axis "mg" is two.

The model needs three SVMs each of which determines whether clauses segmented a test set into by Japanese pause marks are O, D and T respectively. For example, a SVM determining whether a clause is O learns a training set labeled O as true data and learns a training set not labeled O as false data. And the SVM finds an optimal hyperplane separating true data and false data. Furthermore, ChaSen segments the clauses of the test set into words and the model puts each clause into the vector space. The SVM determines whether each clause is O or not utilizing the hyperplane.

Three SVMs labeling the clauses of the test set, the clauses have no label, one label or two or three labels. The model erases clauses that have no label because it is not thought to be connected clinical information. Next, the model divides clauses that have two or three labels exploiting TF/IDF.

The model uses the other kind of training sets that is segmented by personal into clauses labeled by personal. ChaSen divides the clauses of the training set into words and the model measures weight vectors of nouns and verbs.

The number of weight vectors' dimensions corresponds a number of classes. In this case, classes are O, D and T and a number of classes is three. The weight vector for word t is shown as follows.

$$V_t = (V_{t,\mathrm{O}}, V_{t,\mathrm{D}}, V_{t,\mathrm{T}}) \tag{5}$$

Value $V_{t,c}$ corresponding class c is found as follows.

$$V_{t,c} = tf(t,c)idf'(t) \tag{6}$$

$tf(t,c)$ is a term frequency of t in c and $idf'(t)$ is modified version of formula 4 and found as follows.

$$idf'(t) = \ln \frac{\mathrm{NoC}}{df'(t)} \tag{7}$$

NoC is a number of classes and is three in this case. $df'(t)$ is a number of classes including t. If only one class includes t, $idf'(t)$ is highest. If all classes include t, $idf'(t)$ is 0 and t is not thought to be available to classify clauses. While $idf(t)$ of formula 4 shows a weight of a word to identify documents, $idf'(t)$ of formula 7 shows a weight of a word to identify classes. Table 1 shows words rankings by values of weight vectors. The model divides clauses of a test set that have two or three labels using found weight vectors of words.

The model identifies order of occurrence of classes in clause S. We assume that S is up to one cycle. ChaSen divides S into words and the model sets the nouns and the verbs of the words in order of occurrence in S. Let t_1, t_2, …, t_m be the nouns and the verbs of the words in order of occurrence in S. Table 2 shows values of weight vectors corresponding t_1, t_2, …, t_m.

Table 1. Words rankings by values of weight vectors

O		D		T	
認める (recognize)	8.82	認める (recognize)	12.85	施行 (do)	21.39
低下 (deconditioning)	7.78	指摘 (assignment)	9.48	目的 (purpose)	10.13
自覚 (awareness)	7.45	異常 (abnormality)	9.14	紹介 (referral)	9.71
した (have be done)	6.20	疑う (doubt (verb))	8.50	開始 (begin)	8.61
られる ({Japanese suffix})	6.08	診断 (diagnosis)	8.48	日当 (daily pay)	8.16
出現 (appearance)	5.41	られる ({Japanese suffix})	8.28	投与 (administration)	7.21
MEQ	5.02	疑い (doubt (noun))	6.46	MG	7.16
頭痛 (headache)	4.68	考える (consider)	6.22	フォロー (follow)	7.06
全身状態 (general status)	4.68	明らか (clearness)	6.08	加療 (treatment)	6.87
改善 (modification)	4.54	伴う (accompany)	5.67	した (have be done)	6.81
腫瘍 (tumor extent)	4.49	障害 (damage)	5.37	受診 (consultation)	6.68
軽快 (recovery)	4.49	SLE	5.07	中止 (discontinuation)	6.34
症状 (sympton)	4.42	可能性 (capability)	4.91	行う (do)	6.02
よう ({Japanese suffix})	4.38	狭心症 (angina)	4.74	変更 (change)	5.89
転倒 (falling)	4.30	内膜 (endomembrane)	4.39	ML	5.77
続く (continue)	4.30	肝障害 (hepatopathy)	4.39	外来 (outpatient clinic)	5.20
食後 (after food)	4.30	LESION	4.39	経過観察 (follow-up)	5.00
消失 (disappearance)	4.30	CANCER	4.39	内服 (administration)	4.89
倦怠感 (fatigue)	4.30	ADENOCARCINOMA	4.39	受ける (consult)	4.71
圧迫感 (feeling of pressure)	4.30	発症 (occurrence)	4.20	センター (center)	4.56

Table 2. Values of weight vectors corresponding t_1, t_2, …, t_m

	$V_{t,\mathrm{O}}$	$V_{t,\mathrm{D}}$	$V_{t,\mathrm{T}}$
t_1	$V_{t_1,\mathrm{O}}$	$V_{t_1,\mathrm{D}}$	$V_{t_1,\mathrm{T}}$
t_2	$V_{t_2,\mathrm{O}}$	$V_{t_2,\mathrm{D}}$	$V_{t_2,\mathrm{T}}$
…	…	…	…
t_m	$V_{t_m,\mathrm{O}}$	$V_{t_m,\mathrm{D}}$	$V_{t_m,\mathrm{T}}$

We assume that S is labeled as O and D. If S has other two labels or three labels, the model can follow the procedure below.

Let $V_{ti,\mathrm{O}}$ be the highest value in $V_{t1,\mathrm{O}}$, $V_{t2,\mathrm{O}}$, …, $V_{tm,\mathrm{O}}$ and $V_{tj,\mathrm{D}}$ be the highest value in $V_{t1,\mathrm{D}}$, $V_{t2,\mathrm{D}}$, …, $V_{tm,\mathrm{D}}$. The model divides S in order of O, D if $i < j$, and the model divides S in order of D, O if $j < i$. If $i = j$, the model compares $V_{ti,\mathrm{O}}$ and $V_{tj,\mathrm{D}}$, and the class of higher value is fixed. We assume that $V_{ti,\mathrm{O}}$ is higher value. The model erases $V_{tj,\mathrm{D}}$ from $V_{t1,\mathrm{D}}$, $V_{t2,\mathrm{D}}$, …, $V_{tm,\mathrm{D}}$, and let $V_{tj',\mathrm{D}}$ be the highest value. The model divides S in order of O, D if $i < j'$, and the model divides S in order of D, O if $j' < i$. If a number of classes is more than a number of words in S, the model regards S as having only labels corresponding words.

Fixing order of classes in S, the model sorts the words into the classes. We assume that $i < j$. The model sorts t_1, t_2, …, t_i into class O and t_j, t_{j+1}, …, t_m into class D. And the model compares $V_{tk,O}$ and $V_{tk,D}$ corresponding t_k ($i < k < j$) and sorts t_{i+1}, t_{i+2}, …, t_{j-1} into a class corresponding higher value.

The model made ODT elements each of which has some nouns and some verbs. However, ODT elements are not in strict order of O, D, T, O, D, T, … . The model modifies ODT elements.

- Continued Os, continued Ds and continued Ts get together.
- If a cycle lacks, for example, T follows O, the model regards D as existing between O and T.

Figure 5 shows modified ODT cycles.

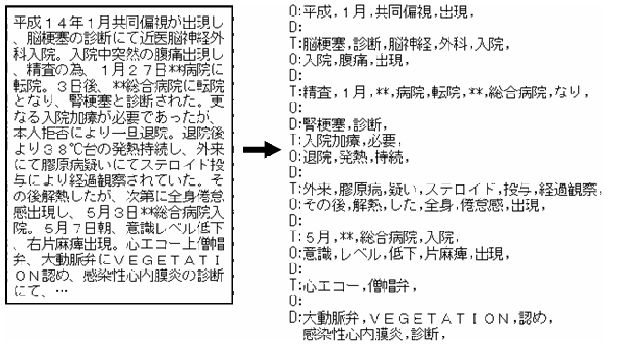

Figure 5. Modified ODT cycles

3.2. Measuring similarities between medical documents

ODT elements among one ODT cycle are related strongly because it is thought that O introduces next D and T, D leads next T and pulls out next O, and similarly T pulls out next O and D. Additionally, we can recognize flows of medical documents focusing Os, Ds or Ts respectively. For example, focusing Ds, we can recognize a flow of diseases such as a patient having cataract after having diabetes. So, the model regards anteroposterior relations among one ODT cycle and anteroposterior relations between Os, Ds or Ts.

Let D_1, D_2 be compared documents. Let X_1, X_2, …, X_{n1} (Y_1, Y_2, …, Y_{n2}) be ODT elements in order of occurrence in D_1 (D_2). The model measures a similarity between D_1 and D_2 as follows.

$$\sum_{1\le i<j\le m1}\ \sum_{1\le k<l\le n2} simi(X_i, X_j, Y_k, Y_l) \qquad (8)$$

The model finds $simi(X_i, X_j, Y_k, Y_l)$ 0 if X_i, X_j, Y_k and Y_l do not meet following two conditions.

1. X_i and X_j are in one ODT cycle and Y_k and Y_l are in one ODT cycle, or X_i and X_j have the same kind of labels and Y_k and Y_l have the same kind of labels.
2. X_i and Y_k have the same kind of labels and X_j and Y_l have the same kind of labels.

If above two conditions are met, the model finds $simi(X_i, X_j, Y_k, Y_l)$ as follows.

$$simi(X_i, X_j, Y_k, Y_l) = \Big(\sum_{c_m\in C(X_i,Y_k)} idf(c_m)\Big)^2 \Big(\sum_{c_n\in C(X_j,Y_l)} idf(c_n)\Big)^2 \qquad (9)$$

$C(X_i, Y_k)$ ($C(X_j, Y_l)$) is a set of words included in both X_i and Y_k (in both X_j and Y_l respectively). $idf(t)$ shows a weight of word t and formula 4 measures it. The less documents include t , the higher $idf(t)$ is.

Finally, the model corrects similarities by dividing them by squared numbers of words of comparative documents because similarities are dependent on sizes of documents. And so, similarities are not absolute values but relative values. A similarity of a medical document D_1 compared to a medical document D_2 is different from a similarity of D_2 compared to D_1.

4. Evaluation

4.1. Retrieval processes and evaluation method

Retrieval processes consist of a matching process, a ranking process and a display process. Retrieving models find documents meeting queries in a matching process, and the models rank the found documents in optimality as results of retrieving in a ranking process. In a display process, the models display the results as users can recognize them easily.

We evaluate the proposed model in a matching process and in a ranking process. The model determines whether each compared document is a similar document in a matching process. The model ranks similar documents in a ranking process.

In an experiment 1, we see whether the model can retrieve medical documents of the same disease as each query document to evaluate the model in matching. In an experiment 2, we see whether the model can rank documents that a doctor determines very similar to each query document near the top of documents.

A model compared to the proposed model is a vector space model that measures a similarity by a number of kinds of words included in two documents. The vector space model regards $idf(t)$ of formula 4 as a weight of word t. The vector space model can consider TF, but the vector space model does not consider TF because the model got better results than the model considering TF.

4.2. Experiment 1

We evaluate the proposed model in a matching process. The proposed model and the vector space model measure similarities between the 100 reports. We evaluate how much percentage of reports that have the same disease as each query report the two models rank in the top 20. We suppose medical documents having the same disease should be matched by binary decision.

Given 100 discharge summaries consist of 5 sets of 20 reports, where each set corresponds to one of 5 diseases: “type 2 diabetes”, “lung cancer”, “angina

pectoris", "uterine fibroid" or "cerebral infarction". Because certain diseases, such as type 2 diabets, may trigger other diseases, we selected reports to make given 5 sets independent each other.

4.3. Experiment 2

We evaluate the proposed model in a ranking process. We choose 22 discharge summaries that have "liver cancer" as a disease name and have a doctor choose up to 3 reports compared to each report as similar documents available for clinical care. A doctor is able to choose no similar document. In this article, we do not make the doctor rank similar documents as he may not able to apply a single firm standard to judge resemblance of each pair of documents. Furthermore, while a doctor determines that document *A* is similar to document *B*, he may determine that *B* is not similar to *A* because *B* includes contents of *A*, but contents of *A* is not important to *B*.

The proposed model and the vector space model measure similarities between the 22 reports and the models get 3, 2 or 1 points if the models rank the similar documents determined by a doctor first, second or third respectively.

5. Results

5.1. Experiment 1

Table 3 shows results of experiment 1.

Table 3. Results of experiment 1

		the vector space model	the proposed model
weights of words	1	66.1%	56.5%
	$idf(t)$	77.6%	72.0%

As a result, the proposed model gave 72.0 % while the vector space model gave 77.6 %. There was not much difference between the models and we saw that a number of kinds of words included in compared documents is important to retrieve documents that have the same disease name.

5.2. Experiment 2

A doctor determined that 18 reports of 22 have similar documents in the experiment 2 and a total number of similar documents is 44.

Table 4 shows similarities compared to medical document 2 and table 5 shows similarities compared to medical document 10.

Numbers of the first columns are numbers of compared medical documents and numbers of the second columns and the third columns are similarities given by the vector space model and the proposed model respectively. Similarities were highest when compared documents were query documents. The doctor determined that documents 9, 10 and 22 are similar documents to the document 2 and documents 2, 8 and 9 are similar documents to the document 10.

Table 4. Similarities compared to medical document 2

	the vector space model	the proposed model
1	0.539	0.005
2	2.576	211.710
3	0.246	0.001
4	0.327	0.014
5	0.304	0.001
6	0.511	0.001
7	0.330	0.001
8	0.460	0.088
9	0.282	0.669
10	0.372	0.297
11	0.216	0.000
12	0.257	0.002
13	0.233	0.285
14	0.292	0.003
15	0.452	0.004
16	0.468	0.001
17	0.390	0.015
18	0.582	0.047
19	0.187	0.000
20	0.323	0.012
21	0.605	0.001
22	0.253	0.003

Table 5. Similarities compared to medical document 6

	the vector space model	the proposed model
1	0.313	0.015
2	0.467	0.468
3	0.237	0.024
4	0.389	0.181
5	0.352	0.060
6	0.627	0.065
7	0.259	0.014
8	0.773	6.565
9	0.299	2.691
10	2.146	415.295
11	0.266	0.020
12	0.206	0.053
13	0.254	0.031
14	0.383	0.208
15	0.334	0.279
16	0.358	0.002
17	0.298	1.411
18	0.519	2.714
19	0.237	0.001
20	0.299	0.316
21	0.576	0.064
22	0.181	0.154

As a result of the experiment 2, the proposed model gave 36 points while the vector space model gave 22 points, and the proposed model could rank similar documents first, second or third for 13 documents (72.2% (13/18)).

6. Discussion

While the proposed model is not thought to be superior to the vector space model in a matching process, the proposed model is thought to be superior to the vector space model in a ranking process. The proposed model gave a good result in a ranking process. It shows that pursuing clinical processes is important to similar document retrieval for medical documents and the proposed model could pursue clinical processes unlike the vector space model and other models that measure a vector of each document.

To construct similar document retrieval system for medical documents, the vector space model finds similar documents in a matching process because the model is executed fast, and the proposed model ranks found similar documents.

Additionally Japanese different words having the same meaning were found in documents. The models do not consider the words, and because of it, the models were thought not to give good results in any cases. The models need structured medical terminology and need to understand relationships between medical terms.

7. Conclusion

We proposed the model that classifies clauses of medical documents exploiting SVMs and TF/IDF into O, D and T and measures similarities between medical documents utilizing the classes. We evaluated the proposed model compared to a vector space model used generally and concluded the proposed model is superior to the vector space model in a ranking process, which is one of retrieval processes. We showed feasibility of similar document retrieval system for medical documents that is executed fast and has high precision when the proposed model is adapted with the vector space model.

8. References

[1] G. Salton and C. Buckley, "Improving Retrieval Performance by Relevance Feedback", *Journal of the American Society for Information Science*, Vol. 41, No. 4, pp. 288-97 (1990).

[2] F. Ingo, T. Ulrich and K. Thomas, "Annotation-based Document Retrieval with Four-Valued Probabilistic Datalog", *Proc. of the first SIGIR Workshop on the Integration Retrieval and Databases*, SIGIR, Sheffield, pp. 31-38 (2004).

[3] J. H. Van Bemmel and J. H. Musen, *Handbook of Medical Informatics*, Springer-Verlag, the Netherlands, pp. 4-7 (1997).

[4] M. Benkhalifa, A. Mouradi and H. Bouyakhf, "Integrating External Knowledge to Supplement Training Data in Semi-Supervised Learning for Text Categorization", *Information Retrieval*, Vol. 4, Issue 2, pp. 91-113 (2001).

[5] J. Guo, A. Takada, K. Tanaka, J. Sato, M. Suzuki, T. Suzuki, Y. Nakashima, K. Araki and H. Yoshihara, "The development of MML (Medical Markup Language) version 3.0 as a medical document exchange format for HL7 messages", *Journal of Medical Systems*, Vol. 28, No. 6, pp. 523-33 (2004).

[6] R. H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F. M. Behlen, P. V. Biron, and A. Shabo Shvo, "HL7 Clinical Document Architecture, Release 2," *Journal of the American Medical Informatics Association*, Vol. 13, No. 1, pp. 30-9 (2005).

[7] T. Takemura, H. Matsui and N. Ashida, "Systematization of Medical Terminology Knowledge based on Examples," Mobile Communications Technology for Medical Care and Triag MCMT 2002, pp. 92-6 (2002).

[8] BANYU PHARMACEUTICAL CO., "The Merck Manuals (Japanese)", http://merckmanual.banyu.co.jp/ (accessed Apr. 2006).

[9] Y. Mastumoto, "ChaSen's Wiki – Front Page", http://chasen.aist-nara.ac.jp/hiki/chasen/ (accessed Apr.).

[10] B. E. Boser, I. M. Guyon and V. N. Vapnik, "A Training Algorithm for Optimal Margin Classifiers", *Proc. of the 5th Annual ACM Workshop on Computational Learning Theory*, pp. 144-152 (1992).

[11] T. Kudo, "Tiny SVM: Support Vector Machine", http://www.chasen.org/~taku/software/TinySVM/ (accessed Apr. 2006).

[12] H. P. Luhn, "A Statistical Approach to Mechanized Encoding and Searching of Literary Information", *IBM Journal of Research and Development*, Vol. 1, No. 4, pp. 307-17 (1957).

[13] J. K. Sparck, "A Statistical Interpretation of Term Specificity and Its Application in Retrieval", *Journal of Documentation*, Vol. 28, No. 1. pp. 11-21 (1972).