Analysis of Intention in Dialogues Using Category Trees and Its Application to Advertisement Recommendation

Hung-Chi Huang **Ming-Shun Lin** **Hsin-Hsi Chen**
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
{hchuang, mslin}@nlg.csie.ntu.edu.tw; hhchen@ntu.edu.tw

Abstract

We propose an intention analysis system for instant messaging applications. The system adopts Yahoo! directory as category trees, and classifies each dialogue into one of the categories of the directory. Two weighting schemes in information retrieval, i.e., *tf* and *tf-idf*, are considered in our experiments. In addition, we also expand Yahoo! directory with the accompanying HTML files and explore different features such as nouns, verbs, hypernym, hyponym, *etc*. Experiments show that category trees expanded with snippets together with noun features under *tf* scheme achieves a best F-score, 0.86, when only 37.46% of utterances are processed on the average. This methodology is employed to recommend advertisements relevant to the dialogue.

1 Introduction

Instant messaging applications such as Google Talk, Microsoft MSN Messenger, Yahoo Messenger, QQ, and Skype are very popular. In the blooming instant messaging markets, sponsor links and advertisements support the free service. Figure 1 shows an example of sponsor links in instant message applications. They are usually randomly proposed and may be irrelevant to the utterance. Thus, they may not attract users' attentions and have no effects on advertisements. This paper deals with the analysis of intention in the dialogues and the recommendation of relevant sponsor links in an ongoing conversation.

In the related works, Fain and Pedersen (2006) survey sponsored search, suggesting the importance of matching advertising content to user intentions. How to match advertiser content to user queries is an important issue. Yih et al. (2006) aimed at extracting advertisement keywords from the intention on the web pages. However, these works did not address the issues in dialogues.

Figure 1. A Sponsor Link in an IM Application

In conventional dialogue management, how to extract semantic concepts, identify the speech act, and formulate the dialogue state transitions are important tasks. The domain shift is a challenging problem (Lin and Chen, 2004). In instant message applications, more challenging issues have to be tackled. Firstly, the discussing topics of dialogues are diverse. Secondly, the conversation may be quite short, so that the system should be responsive instantly when detecting the intention. Thirdly, the utterance itself can be purely free-style and far beyond the formal grammar. That is, self-defined or symbolic languages may be used in the dialogues. The following shows some example utterances.

James: dud, i c ur foto on Kelly's door~ ^^||
Antony: Orz….kill me pls. ><

An intention detecting system has to extract words from incomplete sentences in dialogues. Fourthly, the system should consider up-to-date terms, instead of just looking up conventional dictionaries.

Capturing the intention in a dialogue and recommending the advertisements before its ending are the goal of this approach. This paper is organized as follows. Section 2 shows an overview of the system architecture. Section 3 discusses the category trees and the weighting functions for identifying the intention. Section 4 presents the experimental results comparing with different uses of the category trees and word features. Section 5 concludes and remarks.

2 System Overview

Fain and Pedersen (2006) outlined six basic elements for sponsored search. They are shown as follows:

(1) advertiser-provided content,
(2) advertiser-provided bids,
(3) ensuring that advertiser content is relevant to the target keyword,
(4) matching advertiser content to user queries,
(5) displaying advertiser content in some rank order,
(6) gathering data, metering clicks and charging advertisers.

In instant messaging applications, a dialogue is composed of several utterances issuing by at least two users. They are different from sponsored search in that advertiser content is matched to user utterances instead of user queries. While reading users' conversation, an intention detecting system recommends suitable advertiser information at a suitable time. The time of the recommendation and the effect of advertisement have a strong relationship. The earlier the correct recommendation is, the larger the effect is.

However, time and accuracy are trade-off. At the earlier stages of a dialogue, the system may have deficient information to predict suitable advertisement. Thus, a false advertisement may be proposed. On the other hand, the system may have enough information at the later stages. However, users may complete their talk at any time in this case, so the advertisement effect may be lowered.

Figure 2 shows architecture of our system. In each round of the conversation, we retrieve an utterance from a given instant message application. Then, we parse the utterance and try to predict intention of the dialogue based on current and previous utterances, and consult the advertisement databases that provide sponsor links accordingly. If the information in the utterances is enough for prediction, then several candidates are proposed. Finally, based on predefined criteria, the best candidate is selected and proposed to the IM application as the sponsor link in Figure 1.

In the following sections, we will explore when to make sure the intention of a dialogue with confidence and to propose suitable recommendations. In addition, we will also discuss what word features (called *cue words* hereafter) in the utterances are useful for the intention determination. We assume sponsor links or advertisements are adjunct on the given category trees.

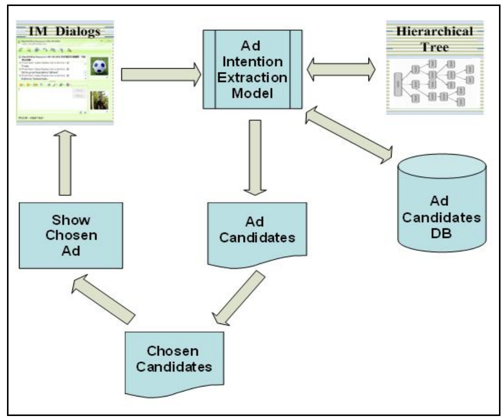

Figure 2. System Architecture

3 Categorization of Dialogues

3.1 Web Directory Used for Categorization

We employ Yahoo! directory[1] to assign a dialogue or part of a dialogue in category representing its intention. Every word in dialogues is classified by the directory. For example, by searching the term *BMW*, we could retrieve the category path:

>Business and Economy>… Makers>Vehicles

Each category contains subcategories, which include some subsidiary categories. Therefore, we could take the directory as a hierarchical tree for searching the intention. Moreover, each node of the tree has attributes from the node itself and its ancestors. Our idea is to summarize all intentions from words in a dialog, and then conclude the intention accordingly.

The nodes sometimes are overlapped, that is, one node could be found in more than one path. For example, the car maker *BMW* has at least two other nodes:

[1] http://dir.yahoo.com

>Regional>Countries>Germany>Business and Economy>…>Dealers

>Recreation>Automotive>…Clubs and Organizations>BMW Car Club of America

The categories of *BMW* include *Business and Economy*, *Regional*, and *Recreation*. This demonstrates the nature of the word ambiguity, and is challenging when the system identifies the intention embedded in the dialogs.

The downloaded Yahoo! directory brings up HTML documents with three basic elements, including titles, links and snippet as shown in Figure 3. The following takes the three elements from a popular site as an example.

Title: *The White House*
Link: *www.WhiteHouse.gov*
Snippet: *Features statements and press releases by President George W. Bush as well…*

Figure 3. Sample HTML in Yahoo! Directory Tree

We will explore different ways to use the three elements during intention identification. Table 1 shows different models and total nodes. YahooO and YahooX are two extreme cases. The former employs the original category tree, while the latter expands the category tree with titles, links and snippets. Thus, the former contains 7,839 nodes and the latter 78,519 nodes.

Tree	Scenario	Total
YahooO	Original Yahoo! directory	7839
YahooX	Yahoo! directory expanded with web title, link, and snippet	78519
YahooS	Yahoo! directory expanded with web snippet	32476
YahooT	Yahoo! directory expanded with web title	39207
YahooL	Yahoo! directory expanded with web link	19690

Table 1. Tree Expansion Scenarios

Tree	Example
YahooT	..\Computers_and_Internet\~W3C HTML Validation Service ..\News_and_Media\~YouTube ..\Science\~Scientific American ..\Social_Science\~Amnesty International ..\Society_and_Culture\~MySpace ..Recreation\Gambling\~Mahjong
YahooS	..\Arts\Artists\Masters\~ Yann Arthus-Bertrand is a photojournalist specializing in adventure, sports, and nature photography. His best know works are aerial shots taken from a hot air balloon. ..\Business_and_Economy\Shopping_and_Services\ ~AltaVista provides web and newsgroup search engines, as well as paid submission services ..\Computers_and_Internet\Desktop_Publishing\~ Converts Mac Word files to Quark Xpress
YahooL	..\ Computers_and_Internet\~www.java.sun.com ..\Health\~www.webmd.com ..\Education\Programs\~www.sea.edu ..\Entertainment\~www.pandora.com ..\News_and_Media\~www.nytimes.com ..\News_and_Media\~www.youtube.com

Table 2. Examples of Expanded Nodes

Table 2 lists some examples to demonstrate the category tree expansion. Some words inside the three elements rarely appear in dictionaries or encyclopedias. Thus, we can summarize these trees and build a new dictionary with definitions. For example, we could find the hottest web sites ***YouTube*** and ***MySpace***, and even the most popular Chinese gamble game, ***Mahjong***.

3.2 Scoring Functions for Categorization

Given a fragment F of a dialogue, which is composed of utterances reading up to now, Formula 1 determines the intention I_{INT} of F by counting total scores of *cue words* w in F contributing to I.

$$I_{INT} = \arg\max \sum_{w \in F} tf(w) \times b(w, I) \quad (1)$$

where $tf(w)$ is term frequency of w in F, and $b(w,I)$ is 1 when w is in the paths corresponding to the intention I_{INT}; $b(w,I)$ is 0 otherwise.

Formula 2 considers the discriminating capability of each cue word. It is similar to *tf-idf* scheme in information retrieval.

$$I_{INT} = \underset{I}{\arg\max} \sum_{w \in F} tf(w) \times \log \frac{N}{df(w)} \times b(w, I) \quad (2)$$

where N is total number of intentions, and $df(w)$ is total intentions in which w appears.

3.3 Features of Cue Words

The features of possible cue words including nouns, verbs, stop-words, word length, hypernym, hyponym, and synonym are summarized in Table 3 with explanation and examples.

Cue word	Explanation	Source	Sample
Noun	Words defined as nouns	Dictionary	Buddha, Earth
Verb	Words defined as verbs	Dictionary	Go, get, replace
Stopword	Words used often but not meaningful	Dictionary	Of, on, off, alongside
Length	Total number of letter in a word	Calculation	Length of "on" is two
Hypernym	A superclass of a word	WordNet	Dog is a hypernym of puppy
Hyponym	A subclass of a word	WordNet	Puppy is a hyponym of dog
Synonym	Same meaning words	WordNet	Feline is a synonym of cat

Table 3. Cue Words Explored

Nouns and verbs form skeletons of concepts are important cues for similarity measures (Chen et al., 2003), so that they are considered as features in our model. Word length is used to filter out some unnecessary words because the shorter the word is, the less meaningful the word might be. Here we postulate that instant messaging users are not willing to type long terms if unnecessary.

In this paper, we regard words in an utterance of dialogues as query terms. Rosie et al. (2006) showed that query substitution may be helpful to retrieve more meaningful results. Here, we use hypernym, hyponym and synonym specified in WordNet (Fellbaum, 1998) to expand the original utterance.

3.4 Candidate Recommendation

The proposed model also provides the ability to show the related advertisements after intention is confirmed. As discussed, for each of node in the category tree, there is an accompanying HTML file to show some related web sites and even sponsors. Therefore, we can also use the category tree to put sponsor links into the HTML files, and just fetch the sponsor links from the HTML file on the node to the customers.

The algorithm to select the suitable candidates could be shortly described as the Longest Path First. Once we select the category of the intention, the nodes appearing in the chosen category will then be collected into a set. We will check the longest path and provide the sponsor links from the node.

4 Experimental Results

4.1 Performance of Different Models

To prepare the experimental materials, we collected 50 real dialogs from end-users, and asked annotators to tag the 50 dialogs with 14 given Yahoo! directory categories shown in Table 4. Average number of sentences is 12.38 and average number of words is 56.04 in each dialog. We compare the system output with the answer keys, and compute precision, recall, and F-score for each method.

Category	Abbreviation
Arts & Humanities	A
Business & Economy	B
Computers & Internet	C
Education	D
Entertainment	E
Government	F
Health	G
News & Media	H
Recreation & Sports	I
Reference	J
Regional	K
Science	L
Social Science	M
Society & Culture	N
na	X

Table 4. Category Abbreviation

Table 5 shows the performance of using Formula 1 (i.e., *tf* scheme). This model is a combination of a scenario shown in Table 1 and features shown in Table 3. For example, the YahooS-noun matches cue words of POS *noun* from utterances to the category tree expanded with *snippets*. WL denotes word length. Only cue words of length ≥ WL is considered. C denotes the number of dialogues correctly analyzed. NA denotes the number of undecidable dialogues. P, R and F denote precision, recall and F-score.

Table 5 shows that YahooS with noun features achieves a best performance. Noun feature works impressively well with the orders, YahooS, YahooT, YahooX, and YahooL. That meets our expectation because the information from snippets is well enough and does not bring in noise as the YahooX. YahooT, however, has good but insufficient information, while YahooL is only suitable for dialogs directly related to links.

Moreover, the experimental results show that verb is not a good feature no matter whether the category tree is expanded or not. Although some verbs can explicitly point out the intention of dialogues, such as *buy*, *sell*, *purchase*, *etc*, the lack of verbs in Yahoo! directory makes the verb features less useful in the experiments. Table 6 shows the performance of using Formula 2 (i.e., *tf-idf* scheme). The original category tree with hyponym achieves the best performance, i.e., 56.56%. However, it cannot compete with most of models with *tf* scheme.

Model	WL	C	NA	P	R	F
YahooS noun	0	43	0	86	86	86.00
YahooT noun	3	38	0	76	76	76.00
YahooX noun	3	37	0	74	74	74.00
YahooL noun	3	37	0	74	74	74.00
YahooS no-stopword	1	36	0	72	72	72.00
YahooO noun	3	34	1	69	68	68.69
YahooS hypernym	4	34	0	68	68	68.00
YahooS	4	33	0	66	66	66.00
YahooT	4	33	0	66	66	66.00
YahooX no-stopword	4	33	0	66	66	66.00
YahooT no-stopword	4	33	0	66	66	66.00
YahooX synonym	4	33	0	66	66	66.00
YahooX	4	32	0	64	64	64.00
YahooO noun verb	0	31	1	63	62	62.63
YahooO	3	31	1	63	62	62.63
YahooS hyponym	4	31	0	62	62	62.00
YahooL	4	31	0	62	62	62.00
YahooL no-stopword	4	30	0	60	60	60.00
YahooT synonym	3	29	0	58	58	58.00
YahooX hyponym	4	29	0	58	58	58.00
YahooL synonym	4	29	0	58	58	58.00
YahooT hyponym	4	28	0	56	56	56.00
YahooO hyponym	3	27	0	54	54	54.00
YahooL hyponym	4	26	0	52	52	52.00
YahooO hypernym	3	25	0	50	50	50.00
YahooT hypernym	5	23	0	46	46	46.00
YahooS verb hyponym	1	17	1	35	34	34.34
YahooS verb	1	15	1	31	30	30.30
YahooO verb	4	2	42	25	4	6.90

Table 5. Performance of Models with *tf* Scheme

Model	WL	C	NA	P	R	F
YahooO hyponym	3	28	1	57	56	56.56
YahooO no-stopword	3	28	1	57	56	56.56
YahooO	3	27	1	55	54	54.54
YahooO hypernym	3	26	1	53	52	52.52
YahooO noun	3	25	1	51	50	50.50
YahooL noun	4	18	0	36	36	36.00
YahooT no-stopword	3	17	0	34	34	34.00
YahooL	4	17	0	34	34	34.00
YahooT noun	3	17	0	34	34	34.00
YahooT noun, hyponym, hypernym	3	16	0	32	32	32.00
YahooT	3	15	0	30	30	30.00
YahooX noun	2	11	0	22	22	22.00
YahooS noun	3	10	0	20	20	20.00
YahooS no-stopword	4	9	0	18	18	18.00
YahooO verb	3	6	32	33	12	17.65
YahooS verb	4	7	18	22	14	17.07
YahooX	1	8	0	16	16	16.00
YahooX no-stopword	1	8	0	16	16	16.00
YahooS	4	7	0	14	14	14.00

Table 6. Performance of Models with *tf-idf* Scheme

4.2 Hit Speed

Besides precision, recall and F-score, we are also interested if the system captures the intention of the dialogue at better timing. We define one more metric called *hit speed* in Formula (3). It represents how fast the sponsor links could be correctly suggested during the progress of conversations. For each utterance in a dialogue, we mark either *X* or a predicted category. Here *X* denotes *undecidable*.

Assume we have a dialogue of 7 utterances and consider the following scenario. At first, our system could not propose any candidates in the first two utterances. Then, it decides the third and the fourth utterances are talking about *Business and Economy*. Finally, it determines the intention of the dialogue is *Computer and Internet* after reading the next three utterances. In this example, we get an *answer string*, *XXBBCCC*, based on the notations shown in Table 4. If the intention annotated by human is *Computer and Internet*, then the system starts proposing a correct intention from the 5^{th} utterance. In other words, the information in the first 4 utterances is not sufficient to make any decision or make wrong decision.

Let *CPL* be the length of correct postfix of an answer string, e.g., 3, and *N* be total utterances in a dialogue, e.g., 7. *HitSpeed* is defined as follows.

$$HitSpeed = \frac{CPL}{N} \tag{3}$$

In this case, the *hit speed* of intention identification is 3/7. Intuitively, our goal is to get the hit speed as high as possible. The sooner we get the correct intention, the better the recommendation effect is.

The average hit speed is defined by Formulas (4) and (5). The former considers only the correct dialogues, and the latter considers all the dialogues. Let *M* and *N* denote total dialogues and total correct dialogues, respectively.

$$AvgHitSpeed = \frac{\sum_{i=1}^{M} HitSpeed_i}{N} \tag{4}$$

$$AvgHitSpeed = \frac{\sum_{i=1}^{M} HitSpeed_i}{M} \tag{5}$$

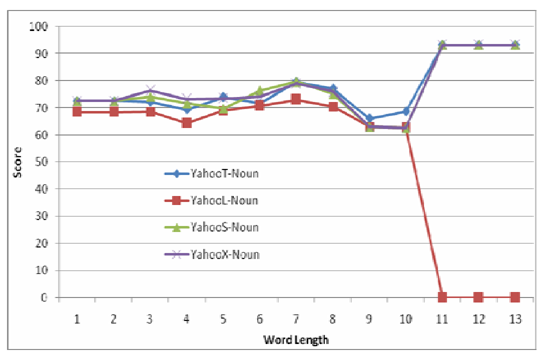

Figure 4. Average Hit Speed by Formula (4)

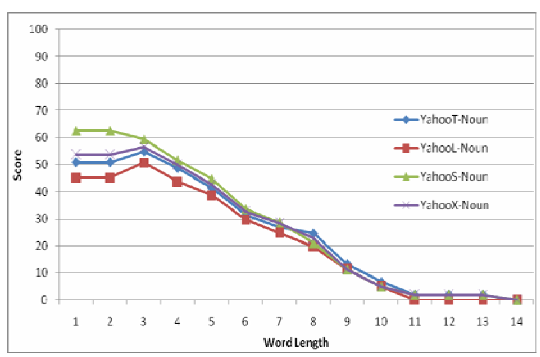

Figure 5. Average Hit Speed by Formula (5)

Figures 4 and 5 demonstrate average hit speeds computed by Formulas (4) and (5), respectively. Here four leading models shown in Table 5 are adopted and nouns are regarded as cue words. Figure 4 shows that the average hit speed in correctly answered dialogues is around 70%. It means these models can correctly answer the intention when a dialogue still has 70% to go in the set of correctly answered dialogs.

Figure 5 considers all the dialogues no matter whether their intentions are identified correctly or not. We can still capture the intention with the hit speed 62.54% for the best model, i.e., YahooS-noun.

5 Concluding Remarks

This paper captures intention in dialogues of instant messaging applications. A web directory such as Yahoo! directory is considered as a category tree. Two schemes, revised *tf* and *tf-idf*, are employed to classify the utterances in dialogues. The experiments show that the *tf* scheme using the category tree expanded with snippets together with noun features achieves the best F-score, 0.86. The hit speed evaluation tells us the system can start making good decision when near only 37.46% of total utterances are processed. In other words, the recommended advertisements can be placed to attract users' attentions in the rest 62.54% of total utterances.

Though the best model in the experiments is to use nouns as features, we note that another important language feature, verbs, is not helpful due to the characteristic of the category tree we adopted, that is, the absence of verbs in Yahoo! directory. If some other data sources can provide the cue information, verbs may be taken as useful features to boost the performance.

In this paper, only one intention is assigned to the utterances. However, there may be many participants involving in a conversation, and the topics they are talking about in a dialogue may be more than one. For example, two couples are discussing a trip schedule together. After the topic is finished, they may continue the conversation for selection of hotels and buying funds separately in the same instant messaging dialogue. In this case, our system only decides the intention is *Recreation*, but not including *Business & Economy*.

Long time delay of response is another interesting topic for instant messaging dialogues. Sometimes one participant could send a message, but have to wait for minutes or even hours to get response. Because the receiver might be absent, busy or just off-line, the system should be capable of waiting such a long time delay before a complete dialogue is finished in practical applications.

Opinion mining is also important to the proposed model. For example, dialogue participants may talk about buying digital cameras, and one of them has negative opinions on some products. In such a case, an intelligent recommendation system should not promote such products. Once opinion extraction is introduced to intention analysis systems, customers can get not only the conversation-related, but also personally preferred sponsor links.

Acknowledgments

Research of this paper was partially supported by Excellent Research Projects of National Taiwan University, under the contract 95R0062-AE00-02.

References

H.H. Chen, J.J. Kuo, S.J. Huang, C.J. Lin and H.C. Wung. 2003. A Summarization System for Chinese News from Multiple Sources. *Journal of American Society for Information Science and Technology*, 54(13), pp. 1224-1236.

D. C. Fain and J. O. Pedersen. 2006. Sponsored Search: A Brief History. *Bulletin of the American Society for Information Science and Technology*, January.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press.

R. Jones, B. Rey, O. Madani, and W. Greiner. 2006. Generating Query Substitutions. In *Proceedings of the 15th International Conference on World Wide Web*, 2006, pp. 387-396.

K.K. Lin and H.H. Chen. 2004. Extracting Domain Knowledge for Dialogue Model Adaptation. In *Proceedings of 5th International Conference on Intelligent Text Processing and Computational Linguistics*, Lecture Notes in Computer Science, LNCS 2945, Springer-Verlag, pp. 70-78.

W. Yih, J. Goodman, and V. R. Carvalho. 2006. Finding Advertising Keywords on Web Pages. In *Proceedings of the 15th International Conference on World Wide Web*, pp. 213-222.