Association Pattern Language Modeling

Jen-Tzung Chien, Senior Member, IEEE

Abstract—Statistical $n$-gram language modeling is popular for speech recognition and many other applications. The conventional $n$-gram suffers from the insufficiency of modeling long-distance language dependencies. This paper presents a novel approach focusing on mining long distance word associations and incorporating these features into language models based on linear interpolation and maximum entropy (ME) principles. We highlight the discovery of the associations of multiple distant words from training corpus. A mining algorithm is exploited to recursively merge the frequent word subsets and efficiently construct the set of association patterns. By combining the features of association patterns into $n$-gram models, the association pattern $n$-grams are estimated with a special realization to trigger pair $n$-gram where only the associations of two distant words are considered. In the experiments on Chinese language modeling, we find that the incorporation of association patterns significantly reduces the perplexities of $n$-gram models. The incorporation using ME outperforms that using linear interpolation. Association pattern $n$-gram is superior to trigger pair $n$-gram. The perplexities are further reduced using more association steps. Further, the proposed association pattern $n$-grams are not only able to elevate document classification accuracies but also improve speech recognition rates.

Index Terms—Association pattern, data mining, language model, long distance association, maximum entropy and trigger pair.

I. INTRODUCTION

THERE is no doubt that the statistical language models using $n$-grams play a decisive role in natural language processing. Many applications have been developed, e.g., speech recognition, document classification, information retrieval, optical character recognition, machine translation, writing correction, and bio-informatics [20]. For the application of speech recognition, we aim at searching the most likely word sequence $\hat{W}$, $\hat{W} = \hat{w}_1^{T} = \{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_T\}$ from the observed speech frames $X$ via Bayes decision rule

$$\hat{W} = \arg \max_W p(W|X) = \arg \max_W p_A(X|W)p(W).$$

The recognition hinges on the likelihood of $X$ given the acoustic models $A$ and the prior probability of $W$ known as the language model $p(W)$. For the application of document classification, we find the best category $\hat{c}$ for an observed document with word sequence $W$ such that the posterior probability $p(c|W)$ is maximized

$$\hat{c} = \arg \max_c p(c|W) = \arg \max_c p(W|c)p(c).$$  \hspace{1cm} (2)

Given the domain-specific language models $p(W|c)$, Bayesian document classification system is established. Language modeling has been attracting many researchers in information retrieval community [18]. Basically, $n$-gram model is to determine the probability of word sequence

$$p(W) = \prod_{i=1}^{T} p(w_i|w_{i-1}, \ldots, w_{i-n+1})$$ \hspace{1cm} (3)

where each word $w_i$ depends on historically observed $n - 1$ words $w_{i-n+1} = \{w_{i-1}, \ldots, w_{i-n+1}\}$. Usually, the value $n$ is restricted to 3 to obtain trigram model. In (3), the conditional probability $p(w_i|w_{i-n+1})$ can be calculated via maximum likelihood estimate by counting the number of co-occurrence of words $w_i$ and dividing it by that of $w_{i-n+1}$. Also, we can employ the maximum entropy (ME) estimate [14], which adopts a generalized iterative scaling (GIS) algorithm [9] to calculate the conditional probability in a form of exponential model [17], [19]. In [22], the latent ME was employed to incorporate explicit and implicit features for semantic language modeling. Generally, the performance of language models can be evaluated by speech recognition rate and document classification accuracy. More popularly, the perplexity is used to measure the average word branching factor of language models. When language models were estimated from training corpus, we apply new test data $\{w_1, w_2, \ldots, w_T\}$ and approximate the entropy by

$$H = -\lim_{T \to \infty} \frac{1}{T} \left\{ \sum_W p(W) \log p(W) \right\} \approx -\frac{1}{T} \log p(w_1^T) = -\frac{1}{T} \sum_{i=1}^{T} \log p(w_i|w_{i-n+1}^{i-1}).$$  \hspace{1cm} (4)

Perplexity is defined by $PP = 2^H$. The lower the perplexity is, the smaller the source uncertainty is to achieve better language modeling. Other performance measure was reported to attain good correlation with speech recognition rate [12].

In standard $n$-gram models, there are several weaknesses [6] to be overcome to improve modeling performance. One is the data sparseness problem that can be resolved using smoothing algorithms [3], [23]. The other weakness is due to the domain mismatch between training and test data, which can be tackled through language model adaptation [15], [17]. Here, we focus on dealing with the insufficiency of $n$-gram
models only considering the dependencies of neighboring words. We are finding solutions to modeling long distance word dependencies. In the literature, Bellegarda [2] exploited the latent semantic information (LSI) language model, which unraveled the large-span semantic relationships between words and documents [7]. In [13], the topic-dependent mixtures of language models were presented to capture topic dependencies of words within and across sentences. The topic-dependent ME language model was also built [24]. Furthermore, a long distance bigram was developed via combining distance bigrams of two words with different distances [10]. The distance bigram could be established under ME framework [21]. More attractively, Rosenfeld [19] extended the concept of distance bigram and exploited the long distance trigger pairs as information sources for ME integration. Two distant words with high mutual information were selected as trigger pair. In this study, we start from the trigger pair n-gram and emphasize the discovery of multiword association patterns for long distance language modeling. Our motivations are stimulated by the fact that the associated words appearing in the contexts may not contain only a pair of words but a sequence of word strings. For example, the multiword strings “September 11”, “George Bush” and “Twin Towers” might often come out from the same news document. As illustrated in Fig. 1, trigram models characterize the conditional probability of neighboring words \( p(\text{George} | \text{September}, 11) \) or \( p(\text{Bush} | 11, \text{George}) \). When applying the trigger pair language model [19], [25], the conditional probability of distant word pairs \( p(\text{Bush} | \text{September}) \) or \( p(\text{Towers} | \text{Bush}) \) can be modeled. However, it is unfeasible to incorporate the associations of three distant words together using trigger pair language models. Modeling of long distance word dependencies is restricted. In this paper, we concentrate on exploring the associations of more than two words so as to effectively resolve the insufficient long-distance dependencies in n-gram models [6]. The association patterns of multiple distant words are discovered in training session. In test session, we are able to merge the conditional probability covering two distant words \( p(\text{Bush} | \text{September}, \text{George}) \) or even three distant words \( p(\text{Towers} | \text{September}, 11, \text{Bush}) \). Due to large span of multiple words, the proposed association pattern n-gram allows semantic/discourse knowledge to be included in language models. Importantly, we amend the a priori algorithm [1], which is popular in data mining field, to automatically identify the association patterns. The mutual information of association patterns is measured and adequately contributed to language modeling via the linear interpolation (LI) scheme. Furthermore, we incorporate the contributions of association pattern features using ME framework. LI and ME association pattern n-gram models are examined in the experiments. The proposed language models are evaluated by the metric of perplexity. Also, the performance is validated by the applications of document classification and speech recognition. The remaining of this paper is organized as follows. Next section presents long distance trigger pairs and the approaches to merging trigger pairs into n-gram models. In Section III, we describe LI and ME association pattern n-grams. A mining algorithm is described here to mine for association patterns with multiple associated words. Later on, the experimental setup and databases are mentioned in Section IV. We report a series of experiments to evaluate the effects of different factors in proposed association pattern n-gram. Finally, the conclusions drawn from this paper are given in Section V.

II. TRIGGER PAIR N-GRAM

Although n-gram model is powerful to represent local word dependencies, it is constrained by the fact that it does not model associations longer than n words within or across sentences.
However, important semantic information is usually embedded in long-distance word associations and incorporate their features into language models. In [16], the information retrieval technique was employed to extract long-term context information for n-gram modeling. Also, the trigger pair was chosen as the basic element for extracting information from sequence of historical words [19], [25]. Trigger pair n-gram was exploited to characterize long-distance language dependencies.

A. Selection of Trigger Pairs

In natural language, there exist many associated word pairs, e.g., {sunshine, beach}, {president, country}, etc. If a trigger word \( w_a \) is significantly associated with a future word \( w_b \), the trigger pair \( w_a \rightarrow w_b \) is produced. The key issues of trigger pair n-gram aim at selecting and measuring trigger pairs. In trigger pair selection, we restrict the window size of two associated words to control the number of trigger pairs. When measuring the significance of a trigger pair, it is meaningful to use the average mutual information (AMI) between words \( w_a \) and \( w_b \).

\[
\text{AMI}(w_a; w_b) = p(w_a, w_b) \log \frac{p(w_a | w_b)}{p(w_a)} + p(w_a, \tilde{w}_b) \log \frac{p(w_b | w_a)}{p(w_b)} + p(\tilde{w}_a, w_b) \log \frac{p(w_b | \tilde{w}_a)}{p(w_b)} + p(\tilde{w}_a, \tilde{w}_b) \log \frac{p(\tilde{w}_b | \tilde{w}_a)}{p(\tilde{w}_b)}
\]

(5)

where \( p(w_a, \tilde{w}_b) \) is the probability of occurring \( w_a \) but without \( w_b \) afterward in the window. Here, \( \text{AMI}(w_a; w_b) \) is order dependent. It measures the information provided by \( w_a \) on \( w_b \). A word pair is recognized as a trigger pair when its AMI is high. The set of trigger pairs \( \Omega_{TR} = \{ w_a \rightarrow w_b \} \) is accordingly selected from the training corpus.

B. LI and ME Trigger Pair N-Grams

Assume that a trigger pair \( w_a \rightarrow w_b \) is observed in word sequence \( W = \{ w_1, \ldots, w_n, \ldots, w_T \} \), the conditional probability \( p(w_b | w_a) \) should be considered in calculating the logarithmic probability using unigram models

\[
\log p_{TR}(W) = \log \{ p(w_1) \cdots p(w_a) \cdots p(w_b | w_a) \cdots p(w_T) \} = \log \{ p(w_1) \cdots p(w_a) \cdots p(w_b) \cdots p(w_T) \} + \log \frac{p(w_b | w_a)}{p(w_b)}.
\]

In (6), the second term represents the mutual information (MI)

\[
\text{MI}(w_a \rightarrow w_b) = \log \frac{p(w_b | w_a)}{p(w_b)} = \log \frac{p(w_a, w_b)}{p(w_a)p(w_b)}
\]

which reflects the degree of preference for associations of \( w_a \) and \( w_b \). In practice, there exist several trigger pairs in word sequence \( W \). We may express trigger pair model as

\[
\log p_{TR}(W) = \sum_{i=1}^{T} \log p(w_i) + \sum_{i=1}^{T-1} \sum_{j=i+1}^{T} \text{MI}(w_i \rightarrow w_j)
\]

(8)

where \( \text{US} \) denotes the predefined window size. The mutual information of all possible trigger pairs is appended within the window size. For each word \( w_a \), we only allow merging information from a single trigger pair \( w_a \rightarrow w_b \). To improve the performance, we may combine knowledge sources from the trigger pair model \( p_{TR}(W) \) and the static \( n \)-gram model \( p(W) \). The linear interpolation (LI) trigger pair \( n \)-gram is generated by [25]

\[
\log p_{LI}(W) = \eta \log p_{TR}(W) + (1 - \eta) \log p(W),
\]

(9)

More attractively, we can incorporate the information source of trigger pairs into \( n \)-gram via the ME principle. ME trigger pair \( n \)-gram models with maximal smoothness can be estimated in an exponential form [19]

\[
p_{ME}(w_i | w_{i-n+1}^{i-1}) = \frac{1}{Z_{X}} \exp \left( \sum_{k=1}^{K} \lambda_k f_k(W_{s|i}) \right).
\]

(10)

Normalization constant \( Z_{X} \) is given by

\[
Z_{X} = \sum_{w_i} \exp \left( \sum_{k=1}^{K} \lambda_k f_k(W_{s|i}) \right).
\]

(11)

In (10) and (11), \( \{ f_k(W_{s|i}), k = 1, \ldots, K \} \) is a set of features or constraints, which is formed by the preceding \( i \) words of sentence \( W_s \) and \( \lambda_k \) is a Lagrange parameter for \( f_k(W_{s|i}) \). GIS algorithm is popular for finding \( \lambda_k \) [9]. Basically, each trigger pair \( w_a \rightarrow w_b \) serves as a feature, imposing the constraint

\[
f_k(W_{s|i}) = f_{w_a \rightarrow w_b}(W_{s|i}) = \begin{cases} 1, & \text{if } w_a \in W_{s|i-1}, w_b = w_i \\ 0, & \text{otherwise} \end{cases}
\]

(12)

on ME estimation. By combining the constraints of trigger pairs and conventional \( n \)-grams, ME trigger pair \( n \)-gram model \( p_{TR}(w_i | w_{i-n+1}^{i-1}) \) is constructed. In this study, we endeavor to relax the constraint of trigger pair \( n \)-gram where only the associations of two distant words are retrieved. A novel association pattern language model is presented hereafter.

III. Association Pattern N-GRAM

Starting from the trigger pair \( n \)-gram, we would like to construct association patterns of more than two distant words from training articles and merge their features into \( n \)-gram models. Interestingly, the selection of associated words is similar to the problem of discovering association rules in a large database of sales transactions, which has been extensively discussed in

CHIEN: ASSOCIATION PATTERN LANGUAGE MODELING 1721
data mining community [11]. Such technology enables marketers to develop customized marketing programs and strategies. In this paper, the text database is referred as the basket data for mining association patterns of multiple words [6]. However, the search space of multiple associated words is really huge. We are trying to efficiently identify all semantic patterns consisting of frequently associated words from training data.

A. Mining for Association Patterns

The underlying concept of association pattern selection is to recursively identify frequent word subsets and perform subset unification. In the beginning, we scan the database and build the frequent one-word subset \( L_1 = \{ w_{u_b} \} \). Term frequency of each word is considered for selection. This subset has no association. To explore the frequent two-word subset \( L_2 = \{ w_{u_b} \rightarrow w_{u_b} \} \), we unify different words in \( L_1 \) and generate the candidate two-word subset \( C_2 = \{ w_{u_a} \cup w_{u_b} \} \). Frequent two-word subset is selected from the candidate two-word subset, i.e., \( L_2 \subseteq C_2 \). The selection is based on the average mutual information \( \text{AMI}(w_{u_a}; w_{u_b}) \). We should scan all sentences and check four types of sentence structures so as to calculate \( p(w_{u_a}, w_{u_b}) \), \( p(w_{u_a}, \text{ta}) \), \( p(w_{u_b}, \text{ta}) \) and \( p(\text{ta}, w_{u_b}) \). Those pairs \( w_{u_a} \rightarrow w_{u_b} \) exceeding the minimum AMI form the subset \( L_2 \). We say the association step of \( L_2 \) is one \((d = 1)\) because trigger word \( u_b \) only associates one word \( u_b \). In a general selection procedure, the frequent \( d \)-word subset \( L_d \) is generated from the frequent \((d - 1)\)-word subset \( L_{d - 1} \). Let \( W_{d-1}^{b} \) denote an association pattern in \( L_{d - 1} = \{ W_{d-1}^{b} \} \). We would like to use the sequence of \( d \)-words \( W_{d-1}^{b} \) as the trigger sequence to predict the occurrence of future word \( u_{b} \). Similar to trigger pair \( w_{u_a} \rightarrow w_{u_b} \), the notations \( a \) and \( b \) in \( W_{d-1}^{b} \) and \( u_b \) have the meaning of “triggering” and “triggered”, respectively. The frequent \( d \)-word subset \( L_d = \{ W_{d}^{u_b} \} \) is established by fulfilling the following two passes.

1. **Join pass.** We scan the subset \( L_{d - 1} \) and pick up the pattern \( W_{d-1}^{b} \), where the preceding \( d - 2 \) words are identical to those of \( W_{d-1}^{b} \). The last word \( u_{b} \) of pattern \( W_{d-1}^{b} \) is appended to \( W_{d-1}^{b} \). The unification \( W_{d-1}^{b} \cup u_{b} \) is formed. Accordingly, the candidate \( d \)-word subset \( C_d = \{ W_{d-1}^{b} \cup u_{b} \} \) is generated as shown in Fig. 2.

2. **Prune pass.** Two prune stages are performed. First, we delete the unification \( W_{d-1}^{b} \cup u_{b} \) from \( C_d \) when at least one \((d - 1)\)-word subset of \( d \)-word sequence \( W_{d-1}^{b} \cup u_{b} \) is not in \( L_{d - 1} \). The candidate subset \( C_d \) is therefore refined to \( C_d \subseteq C_d \). To ensure the goodness of selection, we further prune the unification \( W_{d-1}^{b} \cup u_{b} \in C_d \) via evaluating the information evidence using the AMI between trigger sequence \( W_{d-1}^{b} \) and word \( u_{b} \).

\[
\text{AMI}(W_{d-1}^{b}; u_{b}) = p(W_{d-1}^{b}, u_{b}) \log \frac{p(W_{d-1}^{b} | u_{b})}{p(W_{d-1}^{b})} + p(W_{d-1}^{b}, \text{ta}) \log \frac{p(W_{d-1}^{b} | \text{ta})}{p(W_{d-1}^{b})} + p(\text{ta}, u_{b}) \log \frac{p(\text{ta} | u_{b})}{p(\text{ta})} \tag{13}
\]

All sentences \( \{ W_{s} \} \) in training corpus are applied to determine AMI. The qualified association patterns \( \{ W_{d-1}^{b} \cup u_{b} \in C_d \text{AMI}(W_{d-1}^{b}; u_{b}) \geq \text{minimum AMI} \} = \{ W_{d-1}^{b} \cup u_{b} \} \) are finally mined to form the frequent \( d \)-word subset \( L_d \). Certainly, these patterns involve \( d - 1 \) association steps.

To alleviate the search complexity, we preset an upper bound \( d_{\text{up}} \) for the number of words in selected association patterns. The complete association pattern set \( \Omega_{AS} \) covering different association steps are constructed by \( \bigcup_{d = 2}^{d_{\text{up}}} L_d \). Fig. 3 shows the algorithm for mining association patterns. We search the occurrence of association patterns \( W_{d-1}^{a} \rightarrow u_{b} \) within a sentence \( W_{s} \). The influence of an association pattern \( W_{d-1}^{a} \rightarrow u_{b} \) is activated at word \( u_{b} \). The words in association patterns are order dependent and semantically related. These patterns can be referred as the sequential patterns. Having the association pattern set \( \Omega_{AS} \), we merge the features of all association patterns to estimate the association pattern n-gram.

B. LI and ME Association Pattern N-Grams

For comparative study [8], we also carry out association pattern \( n \)-grams using LI and ME frameworks. When linear interpolation is applied, we can combine the mutual information of all association patterns

\[
\text{MI}(W_{d-1}^{a} \rightarrow u_{b}) = \log \frac{p(W_{d-1}^{a} | u_{b})}{p(W_{d-1}^{a})} \tag{14}
\]

into language modeling to yield the association pattern model

\[
\log p_{AS}(W) = \sum_{i=1}^{T} \log p(w_{i}) + \sum_{s=1}^{S} \sum_{j=s}^{S} \text{MI}(W_{d-1}^{s} \rightarrow u_{j}) \tag{15}
\]
We estimate LI association pattern n-gram by combining the association pattern model \( p_{\text{AS}}(W) \) and the static n-gram model \( p(W) \) according to (9). Moreover, it is a natural way to apply ME principle to combine the features of association patterns and conventional n-gram. Each association pattern \( W_{d-1}^{b} \rightarrow w_b \) contributes a feature

\[
 f_{W_{d-1}^{b}}^{W_{d-1}}(W, \delta) = \begin{cases} 
 1, & \text{if } W_{d-1}^{b} \in W_{\infty}^{i}, w_b = w_i \\
 0, & \text{otherwise} 
\end{cases}
\]

(16) for estimation of ME association pattern n-gram \( p_{\text{AP}}(w_i | W_{i-n+1}^{i}) \). This feature is activated when the word \( W_{d-1}^{b} \) is detected in preceding \( i - 1 \) words \( W_{\infty}^{i-1} \). Again, GIS algorithm [9] is applied to find ME parameter \( \lambda_{W_{d-1}^{b}}^{W_{d-1}} \) for the association pattern \( f_{W_{d-1}^{b}}^{W_{d-1}}(W, \delta) \). By substituting the features and ME parameters of association patterns and conventional n-grams, ME association pattern n-gram \( p_{\text{AP}}(w_i | W_{i-n+1}^{i}) \) is calculated according to (10)(11).

C. Example and Illustration

Fig. 4 shows an example of selection process for association patterns via join pass and prune pass. Let the frequent words \( L_1 = \{ \text{Taiwan}, \text{Bush}, \text{President} \} \) be selected from candidate one-word subset \( C_1 = \{ \text{Taiwan}, \text{Bush}, \text{President}, \text{China}, \text{Alliance}, \text{Palestinian}, \text{Israel}, \text{People}, \text{Army} \} \) generated from training articles. After the join pass, we obtain candidate two-word subset \( C_2 \) containing \text{Bush} \cup \text{Taiwan}, \text{President} \cup \text{Taiwan}, \text{Bush} \cup \text{President}, \text{Taiwan} \cup \text{President}, \text{President} \cup \text{Bush} \) and \text{Taiwan} \cup \text{Bush}. If minimum AMI of 0.08 is specified, this subset is pruned to generate frequent two-word subset \( L_2 = \{ \text{Bush} \rightarrow \text{Taiwan}, \text{President} \rightarrow \text{Taiwan}, \text{Bush} \rightarrow \text{President} \} \). When applying the join pass again, we can use \text{Bush} \rightarrow \text{Taiwan} and \text{Bush} \rightarrow \text{President} to produce three-word candidates \text{Bush} \rightarrow \text{President} \cup \text{Taiwan} and \text{Bush} \rightarrow \text{President} \cup \text{Bush}. Here, we cannot produce candidate \text{President}, \text{Taiwan} \cup \text{Bush} because we don’t have two pairs \text{President} \rightarrow \text{Taiwan} and \text{President} \rightarrow \text{Bush} in \( L_2 \) for unification. Furthermore, we find that there exists a two-word subset \( \{ \text{Bush}, \text{President} \} \) of candidate \text{Bush}, \text{President} \cup \text{Taiwan} which is not in \( L_2 \). This candidate is removed at the first prune stage. At the second prune stage, we calculate the AMI of candidate \text{Bush}, \text{President} \cup \text{Taiwan} and finally select the frequent three-word subset \( L_3 = \{ \text{Bush}, \text{President} \rightarrow \text{Taiwan} \} \). The association pattern set is accordingly unified by \( \Omega_{\text{AS}} = L_2 \cup L_3 \).

Different from data mining algorithm [1] developed for transaction database, we propose the algorithm for mining association patterns from text database and apply it for language modeling. In [1] and [11], the association rules \( \{ A \rightarrow B \} \) of sales items exceeding minimum confidence and minimum support were discovered. Confidence is defined by the percentage of transactions containing \( A \) and also containing \( B \). Support measures the percentage of transactions containing the unification \( A \cup B \) in transaction set. In this study, we use information-theoretic AMI for selection of association patterns and ME algorithm for combining these patterns into n-gram models. AMI should be better for measuring dependencies compared to confidence and support because AMI considers four types of occurrences. Importantly, we exploit the association pattern mining algorithm to find association patterns \( \Omega_{\text{AS}} \) consisting of different numbers of associated words \( \{ L_2, L_3, \ldots, L_{\text{asp}} \} \). The proposed association pattern n-gram is a general framework where the mutual information between frequent d-1 word sequence \( W_{d-1}^{b} \) and word \( w_b \) is properly merged. In case of \( d_{\text{wp}} = 2 \), we only build the relationship between trigger word \( w_a \) and associated word \( w_b \). The association patterns contain frequent word pairs \( \{ w_a \rightarrow w_b \} \), i.e., \( \Omega_{\text{AS}} = \Omega_{\text{TR}} = L_2 \). Accordingly, the trigger pair n-gram is referred as a special realization of association-pattern n-gram when \( d_{\text{wp}} = 2 \). In general, the conventional n-gram using bigram and trigram models only characterize the association between neighboring words. Two limitations are imposed. One is the distance of word associations and the other is the number of associated words. Trigger pair n-gram is designed to relax the limitation of association distance. Nevertheless, the association pattern n-gram relaxes not only the limitation of association distance but also that of association word length. Such long-distance modeling is able to implicitly extract semantic knowledge for statistical n-gram modeling. Definitely, the complex association pattern model will suffer from data sparseness problem. It is crucial to learn representative association patterns from limited training data.

IV. EXPERIMENTS

To examine the performance of association pattern n-gram, we conduct a series of experiments for different language models. When implementing trigger pair n-gram and association pattern n-grams, we evaluate LI and ME approaches for combination of information sources. We also investigate the effects of association steps and minimum AMIs when finding association patterns. The proposed methods are evaluated for the applications of Chinese document classification and continuous Mandarin speech recognition.

A. Databases and Implementation Issues

In the experiments, we adopted several databases to build Chinese language systems. Our dictionary was setup from “Sinica corpus” with a size of five million Chinese words. We gathered 32,941 frequent words to form the lexicon. Each word contained at most four Chinese characters. As referred to http://www.sinica.edu.tw/SinicaCorpus, the articles in Sinica corpus were collected from different domains by Institute of Information Science in Academia Sinica, Taiwan. This open source corpus was representative for Chinese language. We used this corpus as training data to estimate n-gram models. The trigram model and long-distance bigram [10] were investigated. The distance of two words was six at most in long-distance bigram. For the application of document classification, we collected 3516 Chinese news documents covering eight categories: Technology, Society, Travel, World, Sports, Entertainment, Politics and Business. These classified documents were sampled from news websites CNA (http://www.cna.com.tw), ChinaTimes (http://news.chinatimes.com) and UDNews (http://www.udnews.com.tw), etc.
1) \( L_d = \{ W_d^d \} = \{ w_j \} \) (frequent words);
2) \( d = 2 \);
3) while \(( L_{d-1} = \{ W_{d-1}^{d-1} \})\) is nonempty and \( d \leq d_{\text{min}} \) do begin
4) apply join pass to generate preliminary candidates \( C_d \).
5) apply first stage of prune pass and refine candidates as \( \tilde{C}_d = \{ W_d^d \cup \{ w_k \} \} \).
6) for all sentences \( W_s \) in training corpus do begin
7) find all candidates \( \{ W_d^d \cup \{ w_k \} \in \tilde{C}_d \} \) contained in \( W_s \).
8) Increment counts for four types of occurrences for these candidates.
9) end
10) Compute AMI(\( W_d^d, w_k \)) for all candidates \( \tilde{C}_d = \{ W_d^d \cup \{ w_k \} \}. \)
11) \( L_d = \{ W_d^d \} = \{ W_d^d \rightarrow w_k \} = \{ W_d^d \cup \{ w_k \} \in \tilde{C}_d | \text{AMI}(\{ W_d^d \cup \{ w_k \} \} \geq \text{minimum AMI} \}; \)
12) \( d = d + 1 \);
13) end
14) \( \Omega_{AS} = \bigcup_{d=2}^{d_{\text{max}}} L_d \);

Fig. 3. Algorithm for mining association patterns.

During the implementation, Witten-Bell smoothing [23] served as the preprocessing step for estimating unseen trigrams in training data. The resulting perplexity was 143.2, which was referred as the baseline result. The perplexity of using long-distance bigram was reduced to 136.5. Using trigger pair or association pattern \( n \)-grams, we set window size to be within a sentence. The interpolation weight of \( \rho \text{TRG}(W)(\bar{p}_{\text{AS}}(W)) \) and static \( n \)-gram \( p(W) \) in (9) was tuned using a held-out test data set containing 398 news documents (April 17). The optimized weight with the lowest perplexity was varied for different association steps. LI \( n \)-grams were realized by adopting these tuned weights, which were found in the range of 0.4 \( \leq \eta \leq 0.8 \). In LI framework, we only allowed the prediction of word \( u_k \) by a single trigger word \( u_{\text{trg}} \) or trigger sequence \( W_d = W_{d-1} \). In ME framework, we did not have this limitation. When implementing GIS algorithm, we used fifteen iterations to compute ME parameters \( \{ \lambda_k, 1 \leq k \leq K \} \). All ME models were established with the constraints of unigrams, bigrams and trigrams. To reduce the processing time of finding association patterns, we used the hash tree data structures in the implementation. We evaluated the relative computation cost and memory requirement for different language models on a personal computer with CPU Pentium III 450 and RAM 256 MB.

For application of speech recognition, we implemented the continuous hidden Markov models (HMMs) to construct acoustic models \( A \) for Mandarin speech, which is known as a syllabic and tonal language. The context dependent modeling of 408 Mandarin syllables was constructed [4], [5]. Currently, the estimated language models \( p(W) \) were translated into syllable language models for syllable decoding of continuous speech. We used one-pass search algorithm and built time-aligned syllable hypothesis lattice according to acoustic HMMs. Syllable language models were then applied for rescoring and finally searching the recognized syllable sequence. Association pattern \( n \)-gram was integrated in syllable decoding while the search complexity was moderate. We reported the syllable recognition rates (%). The benchmark MAT-400 speech database was used to train speaker-independent HMMs. It contained Mandarin speech across Taiwan from 400 speakers (216 males and 184 females) talking over telephone networks [4], [5]. The test set (Test 500) was recorded via telephones and consisted of 500 sentences from 30 outside speakers. It totally included 4754 syllables. All utterances were sampled at 8 kHz with 16-bit resolution. Each frame was characterized by 12 Mel-frequency cepstral coefficients (MFCCs), 12 delta MFCCs, one delta log energy, and one delta-delta log energy. Sentence-based cepstral mean subtraction was applied. The contents of MAT-400 and Test500 were in general domains. Sinica corpus and MAT-400 were adopted to search trigger pairs and association patterns for language modeling.

B. Evaluation of LI and ME Trigger Pair N-Grams

This study focuses on modeling long-distance word dependencies in natural language. The dependencies are tackled through the trigger pair \( n \)-gram, which is viewed as a special case of association pattern \( n \)-gram with single association step. LI and ME approaches are presented to combine information sources of trigger pairs and conventional \( n \)-grams. LI is a suboptimal approach where the mutual information of trigger pairs is merged and interpolated with conventional \( n \)-gram. Using ME approach, trigger pair \( n \)-gram is estimated with least additional assumptions and maximal model smoothness. Trigger pairs serve as the features for knowledge integration. Here, the trigger pair \( n \)-gram is affected by the value of minimum AMI. Minimum AMI is used to determine the size of selected trigger pairs. As shown in Fig. 5, we compare the perplexities of baseline trigram, long-distance bigram, LI trigger pair \( n \)-gram and...
Fig. 4. Example of selection process for association patterns through join pass and prune pass.

Fig. 5. Comparison of perplexities for baseline trigram, long-distance bigram, LI trigger pair $n$-gram and ME trigger pair $n$-gram with different minimum AMI values.

ME trigger pair $n$-gram with various minimum AMI values. We find that LI trigger pair $n$-gram is better than baseline trigram and long-distance bigram for different thresholds. However, ME trigger pair $n$-gram outperforms LI trigger pair $n$-gram. Using ME trigger pair $n$-gram, the lowest perplexity of 124.2 is achieved compared to 129.1 of LI trigger pair $n$-gram. These results are obtained for the case of minimum AMI set to 0.08 while there are 222 trigger pairs merged. In what follows, we roughly preset the minimum AMI to be 0.08 to examine association pattern $n$-gram under different association steps.

C. Evaluation of Association Pattern N-Gram With Different Association Steps

Nevertheless, the novelty of proposed method is to generalize trigger pair $n$-gram to tackle long-distance dependencies with more than one association step. We would like to achieve multiword long-distance language modeling. In implementation of association pattern $n$-gram, we specify the maximal association step and apply the association pattern mining algorithm to search salient patterns covering different association steps. Mutual information of association patterns appearing in articles is merged to build LI association pattern $n$-gram. Also, the features of association patterns and conventional $n$-grams can be integrated through ME estimation of association pattern $n$-gram. A mining algorithm is presented to identify frequent word sets by unifying frequent word subsets and evaluating whether the unified sets meet the criteria. In this manner, the association patterns are recursively extracted from single association step to maximal association step. In Fig. 6, we display...
using LI and ME association patterns, respectively. These results are better than 124.2 of ME trigger pair $n$-gram using only 222 patterns of single association step. The perplexity of ME association pattern $n$-gram is further reduced to 109.3 when using 1438 association patterns with maximal association step being six. Association pattern $n$-gram is superior to trigger pair $n$-gram. It is interesting to see that the perplexity is comparable to that of maximum association step being seven because the patterns with seven association steps are very few in test documents. The contributions of introducing association patterns of more than one association step are obvious. But, the higher association models would be vulnerable to data sparseness problem. We suggest that the proper association step of association pattern $n$-gram should be three or four for real applications.

Further, when evaluating the computation cost in test session, we find that association pattern $n$-gram with four association steps spends extra 7.1% and 5.2% computation times compared to baseline trigram and trigger pair $n$-gram, respectively. These costs are not significant. However, in training session, the computation overheads using association pattern $n$-gram are 45.3% and 31.7% compared to baseline trigram and trigger pair $n$-grams, respectively. It is a tradeoff between perplexity and computation cost. Also, we evaluate the memory requirement of storing language models for baseline trigram and association pattern $n$-gram. We find that the incorporation of association patterns requires extra 14.8% memory cost, which is moderate. However, when maximum association step is increased to six, the training time and memory requirement of association pattern $n$-gram are substantially raised to 87.4% and 20.5% relative to baseline trigram, respectively. In fact, these overheads depend on the length of the training sentences. The longer training sentences induce the higher computation overhead for searching association patterns. Also, it is a good challenge to efficiently discover association patterns across sentences or even in paragraph level. In the subsequent experiments, we only report the results of ME approaches to trigger pair and association pattern $n$-grams. The maximum association step is limited to be four.

D. Applications for Document Classification and Speech Recognition

To see the performance in real-world application, we conduct the experiments of document classification and speech recognition. In document classification, we aim to classify 884 news documents into eight categories. Eight category dependent language models are estimated from training data including Sinica corpus and 2234 classified news documents. Language model approaches are also compared with a standard document classification approach using term frequency—inverse document frequency (TF-IDF). Using TF-IDF, classification system is performed via measuring the cosine similarities between input document vector and eight category vectors. These vectors consist of TF-IDF features of Chinese words appearing in the corresponding documents/categories. To conduct a fair comparison, the term frequencies of unigram, bigram and trigram serve as the features in TF-IDF method. In Fig. 8, we display document classification accuracies (%) for TF-IDF, baseline trigram and association pattern $n$-gram with different maximal association steps. It is obvious that language model approaches attain

---

Fig. 6. Number of association patterns versus association step using association pattern $n$-gram.

Fig. 7. Comparison of perplexities for baseline trigram, LI association pattern $n$-gram and ME association pattern $n$-gram with different maximal association steps.
V. CONCLUSION

This paper investigated the issue of modeling long-distance association in statistical $n$-gram. Long-distance modeling approach using trigger pair $n$-gram was studied. To relax the constraints of trigger pair $n$-gram characterizing only two distant words, this paper explored a novel association pattern $n$-gram where the word associations of frequent word sets consisting of more than two distant words were merged in $n$-gram. The frequent word sets, also called association patterns, were determined through the association pattern mining algorithm. This algorithm recursively performed join step and prune step so as to extend frequent two words to multiple words. The association patterns involving various association steps were identified and unified together. In this study, we consistently used information-theoretic criteria for pattern selection and knowledge integration. The averaged mutual information was applied to judge whether the selected word sets are frequent or not. The maximum entropy criterion was used to optimally combine the features of association patterns in $n$-gram models. Typically, the association pattern $n$-gram was generalized from the trigger pair $n$-gram. In the case of a single association step, association pattern $n$-gram was reduced to trigger pair $n$-gram. In the experiments, we performed association pattern $n$-grams and evaluated them in terms of perplexities, document classification accuracies and speech recognition rates. Our findings are summarized as follows.

1) Trigger pair and association pattern $n$-grams are consistently better than baseline trigram and long-distance bi-gram.

2) Association pattern $n$-gram is superior to trigger pair $n$-gram. The more the association step involves, the better the performance is achieved. However, higher order association models require substantially larger corpora to train properly.

3) Knowledge integration using ME outperforms that using linear interpolation for both trigger pair $n$-gram and association pair $n$-gram.

4) Computation overhead and memory requirement using association pattern $n$-gram are within practical constraints.

5) $N$-gram model based approach is much better than TF-IDF method for document classification. Association pattern $n$-gram obtains good improvement.

6) Incorporation of association pattern $n$-gram does improve speech recognition rate.

In the future, we will investigate the impact of association pattern $n$-gram on a larger speech recognition task. We will develop an objective function to determine AMI threshold for construction of association pattern set. We will also examine the effects of individual trigger pair and association pattern on language modeling.

ACKNOWLEDGMENT

The author acknowledges Dr. Hazen and three anonymous reviewers for providing valuable comments, which considerably improved the quality of this paper.
REFERENCES


Jen-Tzung Chien (S’97–A’98–M’99–SM’04) received the Ph.D. degree in electrical engineering from National Tsing Hua University (NTHU), Hsinchu, Taiwan, R.O.C., in 1997.

He was an Instructor in the Department of Electrical Engineering, NTHU, in 1995. In 1997, he joined the Department of Computer Science and Information Engineering, National Cheng Kung University (NCKU), Tainan, Taiwan. He was a Visiting Researcher at the Speech Technology Laboratory, Panasonic Technologies Inc., Santa Barbara, CA, in the summer of 1998 and the Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan, in the summers of 2002 and 2005. His research interests include general pattern recognition, speech recognition, speaker adaptation, language modeling, information retrieval, face recognition, biometrics, multimedia signal processing, and multimodal human-computer interaction.

Dr. Chien serves on the Board of the International Journal of Speech Technology, the Board of ACLCLP and Chairs the Special Interest Group on Spoken Language Processing. He is a member of the IEEE Signal Processing Society, the International Speech Communication Association, the Chinese Image Processing and Pattern Recognition Society, and the Association for Computational Linguistics and Chinese Language Processing (ACLCLP). He received the Outstanding Young Investigator Award (Ta-You Wu Memorial Award) from National Science Council, Taiwan, in 2003 and the Research Award for Junior Research Investigators from Academia Sinica, Taiwan, in 2004.

Authorized licensed use limited to: National Cheng Kung University. Downloaded on November 4, 2008 at 20:05 from IEEE Xplore. Restrictions apply.