

# *SmartWrite*: Extracting Chinese Lexical Grammar Patterns Using Dependency Parsing

Cheng-Cyuan Peng<sup>1</sup>, Ching-Yu Yang<sup>2</sup>, Jhih-Jie Chen<sup>2</sup>, Jason S. Chang<sup>2</sup>

<sup>1</sup>Institute of Information Systems and Applications  
National Tsing Hua University

<sup>2</sup>Department of Computer Science  
National Tsing Hua University

{zeno, chingyu, jjc, jason}@nlplab.cc

## Abstract

This paper presents an interactive writing prototype system, *SmartWrite*, which provides Chinese word usage in the form of grammar patterns, collocations, and examples to assist language learners in writing. We propose a method of inducing common Chinese grammar patterns from a large-scale Chinese corpus. First, we try to identify grammar patterns based on the dependency relations of a given sentence parsed with existing language parsers. Then, we calculate words in a dependency relation with a target word to automatically generate the collocations. Finally, we find good example sentences that best exemplify each pattern. At run-time, the last content word a user typed is used for obtaining grammar patterns, collocations, and example sentences as continuous writing suggestions. Evaluation on the suggested word usage of real verbs extracted from a Web search engine log shows that the method significantly outperforms underlying search engines, and even prevails over a commercial search engine by providing structural information of verbs.

## 1 Introduction

Many queries are submitted by language learners to online dictionaries or other reference resources everyday, and an increasing number of services on the Web specifically target language learning. For example, VOA special English ([learningenglish.voanews.com](http://learningenglish.voanews.com)) provides grade-level-appropriate news articles for learning how to read and speak, while Corpus of Contemporary American English (COCA) ([corpus.byu.edu/coca/](http://corpus.byu.edu/coca/)) and Corpus of Contemporary Mandarin in Taiwan (COCT) ([coct.naer.edu.tw/cqpweb/](http://coct.naer.edu.tw/cqpweb/)) use an information retrieval system to display collocations and example sentences from a corpus.

Typical linguistic search engines such as *COCA* and *COCT* usually fashion a bottom-up approach by providing a wealth of examples for inductive learning. However, users may be overwhelmed by the examples and are slow to grasp the deep vocabulary knowledge including the grammar patterns of word usage. Such deep lexical knowledge can be acquired more effectively, if a system automatically can induce and display grammar patterns with examples on demand.

Consider the query word “公佈” submitted to find lexical information. The best way to support language learning and assist writing is probably not just providing a random list of collocations and examples related to “公佈”, but rather a list of grammar patterns and corresponding collocations ordered by frequency counts:

- 公佈：V n [結果 名單 報告]  
教育部公佈歐盟獎學金獲獎名單
- 公佈：V clause [為 達]  
經濟部公佈3月外銷訂單為歷史新高

These patterns and collocations can be induced from a very large corpus by parsing and selecting a sample of prototypical sentences transforming their dependency relations into grammar patterns. Intuitively, by selecting a small representative sentences, we can cope with parsing errors and induce a set of reason-

**SmartWrite**

教育部今天公佈

[V] 公佈 n 52721      公佈 [結果, 名單, 報告, 財報, 數據]

下周將公佈結果, 屆時汙染是 ...  
 明年初公佈名單。  
 2010年iF產品設計獎剛剛公佈得獎名單, 臺灣共76項入圍, ...

[V] n 公佈 17134      [名單, 成績, 財報, 結果, 中心] 公佈

林全內閣名單公佈時, 被外界批評為「老藍男」,  
 國中第二次基測成績公佈, 連祐暘總分295分為榜首, 僅錯了社會科一題,  
 今日三星財報公佈之後股價下跌,

[V] 公佈 clause 5080      公佈 [為, 是, 有, 達, 完畢]

去年12月上市櫃公司營收, 今天將公佈完畢, 截至中午前, 有169家公司營收創新高。  
 該公司公佈去年11月單月自結淨損為1.03億元, 而10月單月也約虧損0.7億元,  
 先前公佈以大紅色基調, 改為五彩繽紛的顏色, 設計團隊表示, 修改後的舞臺主要是臺灣紅與藍  
 天綠地為主要色調, 以及多元色彩人民意象, 呈現包容並蓄新政府。

Figure 1: The query “公佈” of *SmartWrite*.

ably accurate lexical grammar patterns and collocations for effective language learning. Moreover, the finding of good examples can bridge the gap between language teaching and learning.

We present a prototype interactive system, *SmartWrite*, that automatically retrieve grammar patterns and collocations for a word. An example of *SmartWrite* writing session is shown in Figure 1. *SmartWrite* displays the most common grammar patterns for the last word in the working area. At run-time, *SmartWrite* presents a working area for the user to write on from scratch or to paste previous written text. *SmartWrite* then detects the last word and retrieves the patterns and collocations for the word in question. In our prototype, *SmartWrite* returns the patterns and collocations to the user directly (see Figure 1); alternatively, the patterns and collocations can be used as input to an automatic essay rater in order to provide grade or corrective feedback.

The rest of this paper is organized as follows. In the next section, we introduce some related work in language learning, lexicography, and existing concordance systems (Section 2). Then, we present our method for automatically extracting collocations

and inducing grammar patterns for language learning (Section 3, Section4). As part of our evaluation, we judge the quality of the patterns, collocations, and examples induced and retrieved by *SmartWrite* in a preliminary evaluation over a set of verbs chosen randomly (Section 5). Finally, we conclude this study in Section 6.

## 2 Related work

Data-Driven Learning (DDL), in particular corpora and concordances, has been an area of active interdisciplinary research for lexicographers, corpus linguists, applied linguists, translators, and computer scientists (Boulton and Cobb (2017), among others). In our work, we address a different aspect of data-driven learning, which was not a direct focus of the concordance developers. We consider the need and method of developing top-down concordance design, where the user is interested in spending less time exploring language usage.

More specifically, we focus on the inducing of grammar patterns to serve as a the structure for a top-down concordance system. For each grammar pattern, more information such as collocations and examples are displayed on demand. In this way, we

can help a second language learner acquire the deep vocabulary knowledge effectively for writing.

Most of the concordance systems used the general approach of using inverted file to retrieve and present sentences. The sentences matching the query word (or phrase) are listed in the form of keyword in context (KWIC) either randomly or ordered by the neighboring words. Learners can discover the patterns of a word from the use of language by themselves (cf. Data-driven learning (Johns, 1991)). However, this approach is not optimal because it encourages a more bottom-up rather than top-down display of concordance lines and the users can only examine one example after another without a structure to follow, making discovery of patterns bottom-up a slow and ineffective learning process. See (Flowerdew, 2009)

Recent concordance systems has begun to organize examples according to the grammatical relations and collocations. Kilgarriff et al. (2014; Kilgarriff et al. (2004) describe an interesting approach for extracting collocations (i.e., word sketches), and to build a concordance and thesaurus. Specifically, they extract grammatical relations of a lexicon using hand-crafted rules, count the frequency of a word which is connected to another word by a specific grammatical relation (e.g., SUBJ OF, OBJ OF) (Lin, 1998), and weigh its collocations by *logDice* (Rychly, 2008).

Also, Hu et al. (2016) presents a system searching for the collocations of a Chinese word with a certain grammar relations, online Chinese Collocation Assistant (COCA) (<http://occa.xingtanlu.cn>). Sentences are preprocessed by LTP-Cloud (Che et al., 2010), including word segmentation, POS tagging and dependency parsing, collocations are extracted based on dependency relations by 37 hand-crafted rules and classified into nine types of grammatical relations. Finally, they try to filter out parsing errors and inappropriate collocations by frequency.

Other systems (e.g., Baisa and Suchomel (2014) and Wu and Witten (2016)) also use collocations as the top-level structure to organize examples. Developing from the traditional system of concordances and collocations, some of the previous researches integrate grammatical relations into collocations (Kilgarriff et al., 2014; Hu et al., 2016) . However, this

approach still limits writing assisting to the lexical level, which is helpful but not enough to describe the whole picture of word usage. To provide more useful assistance for learners, a writing-assisted system should also provide the structural information and the interaction between possible grammatical relations of a word .

In a study more closely related to our work, Yen et al. (2015) proposes an interactive writing environment, *WriteAhead*, which provides the automatically derived English grammar patterns and examples to assist learners to write fluently and avoid making writing errors. The main difference from our current work is that in (Yen et al., 2015), the grammar patterns are derived using all English sentences with contain high degree of parsing errors, while we use the active learning approach to select a small sample of representative Chinese sentences to cope with parsing errors.

In contrast to the previous research in using corpora and concordance for lexicography and language learning, we present a system that automatically extracting accurate grammar patterns, and selecting a sample of representative sentences, with the goal of maximizing the accuracy of inducing and displaying common grammar patterns, frequent collocations, and representative example sentence for effective learning.

### 3 The SmartWrite System

Deriving pedagogical, lexical grammar from a treebank or a dependency parser trained on a tree bank doesn't often work very well. In general, treebanks (or parsers trained on treebanks) cover most syntactic structure of common words, but the amount of the words is not sufficient enough to parse many sentences. Besides, the performance of existing dependency parsers is unstable. When it comes to rare words and complicated syntactic constructions, they often produce erroneous dependency parsings. Underlying poor lexical coverage and erroneous parsings often lead to distortion and noise in subsequent grammar patterns extraction. To induce a set of comprehensive and accurate grammar patterns, a promising approach is to utilize an existing dependency parser and a large-scale corpus to select and process a sample of simple and representative sen-

tences that are less susceptible to parsing errors.

We manage to induce a comprehensive set of common grammar patterns useful for L2 vocabulary learning and assisted writing. These patterns are returned as the output of the system in response to users’ demand. The returned patterns along with collocations and example sentences can be examined by a user directly, or passed on to sophisticated e-raters to produce quality rating and corrective feedback. Thus, it is crucial that the patterns for a content word are comprehensive and accurate. At the same time, the amount of patterns need to not be too large that might overwhelm a user or subsequent language assessment systems. Therefore, our goal is to return a reasonable-sized set of patterns and cover typical usage of a given word.

**Problem Statement:** We are given a general purpose dependency parser *PAR* (e.g., *Stanford Parser*), a large-scale corpus *CORP*, and a content word *W*. Our goal is to induce a set of grammar patterns,  $p_1, \dots, p_k$ , for *W* present in *CORP* using *PAR*. For this, we parse *CORP* using *PAR* in order to select a sample of most representative sentences  $S_1, \dots, S_k$ , consisting of *W*’s most frequent dependencies,  $d_1, \dots, d_n$ , and associated collocates  $c_1, \dots, c_m$ , such that these dependency relations and collocations cover more than  $K$  percent of the usage of *W* in *CORP*.

The following part of this paper moves on to describe our solution to this problem (Section 4). This strategy relies on an existing dependency parser and a corpus. Finally, we show how *SmartWrite* retrieves and displays the patterns, collocations, and example sentences (Section 4.5).

## 4 Grammar Patterns Induction

In this section, we describe our grammar pattern induction process in detail. We try to identify common grammar patterns and collocations of a given word that represent its grammatical behavior. Overview of the inducing process is shown in Figure 2.

### 4.1 Dependency Parsing

In the first stage of the inducing process, we perform *dependency parsing* on each sentence in the *corpus*. An example of dependency parsing result is shown in Table 1. After obtaining the dependency and PoS, we then convert the Penn Chinese Treebank format

| loc | Word | Pos | Relation | Dep. |
|-----|------|-----|----------|------|
| 1   | 政府   | N   | nsubj    | 6    |
| 2   | 卻    | ADV | advmod   | 6    |
| 3   | 還    | ADV | advmod   | 6    |
| 4   | 一直   | ADV | advmod   | 6    |
| 5   | 不    | ADV | neg      | 6    |
| 6   | 公佈   | V   | root     | 0    |
| 7   | 相關   | ADJ | amod     | 8    |
| 8   | 細節   | N   | dobj     | 6    |

Table 1: A example of the dependency parser *Stanford-Parser*.

to a simplified PoS label (e.g., VV to V, NT to N). By simplifying PoS, we reduce the granularity of patterns and thus make the use of patterns easier to understand. For simplicity, we only consider verbs as headword for later pattern induction. Next, we preserve the headword and collocation information of word distances from root, dependencies, PoS (“root” PoS will be capitalized and its collocations will be in lower case). If the PoS of a “root” is Verb, we do not regard children with less important dependencies (e.g., modifiers) as collocates. On the other hand, we consider grandchildren of “root” with “pobj” or “pccomp” dependency label as collocates. Then, we mark the headword and its collocations with “[ ]” brackets (e.g., 公佈 內容, 0 2, root dobj, V n, 但沒有 [公佈] 合約 [內容]). In this step, we will recognize dependencies, collocations, dependency distances, PoS, and “root” words.

### 4.2 Extracting Dependencies and Collocations

In the second stage of the inducing process, we extract verbal dependencies that are not optional modifiers, including:

- Direct dependency of verb: **nsubj, dobj, range, topic, prep, ccomp**
- Dependencies of prep: **pobj, pccomp**
- Dependencies of ccomp: **nsubj, dobj, range, topic, prep**

We compute the count of dependencies and associated collocations and then retain the most frequent dependencies,  $d_1, \dots, d_n$ , and associated collocates  $c_1, \dots, c_m$ . A sample output of this stage is shown in Table 2.

---

**Procedure *ExtractPatterns***

1. Parse sentences in the given Corpus into dependencies (Section 4.1)
  2. Extract most frequent dependencies and associated collocations (Section 4.2)
  3. Select representative sentences consisting of common dependencies and collocations (Section 4.3)
  4. Convert dependencies into grammar patterns (Section 4.4)
  5. Rank and filter patterns (Section 4.5)
- 

Figure 2: The pattern extraction process.

| Dependency | Collocation                       | Count  |
|------------|-----------------------------------|--------|
| V:subj:N   | 公司 名單 美國 教育部 衛生局 中國 中心 結果 政府 財政部  | 94,900 |
| V:obj:N    | 結果 名單 財報 數據 報告 營收 調查 數字 統計 指數     | 88,900 |
| V:在:N      | 網站 網站-上 網絡-上 日 官網 下午 今天 月底 網絡 本   | 9,440  |
| V:clause:V | 結 達 是 值 合併 添加 創 獲利 下滑             | 8,730  |
| V:amount:N | 元 次 美元 年 家 件 季 天 日 周              | 3,600  |
| V:於:N      | 網站 今天 官網 月底 日前 下午 近日 下 本 網站-上     | 3,570  |
| V:對:N      | 外 調查 媒體 大 結果 預測 公佈 外界 調整 馬林魚-後    | 2,340  |
| V:由:N      | 總統 統一 單位 行政院 廣達 交通部 中油 南科 總統府 力晶  | 600    |
| V:以:N      | 方式 形式 選委會 元 新聞稿 主計處 消息 行政命令 公司 網絡 | 412    |
| V:向:N      | 社會 外界 媒體 大 全世界 國人 公 大家 國民 民       | 336    |

Table 2: Example patterns extracted for the headword “公佈”

### 4.3 Filtering Sentences

In the third stage, for each word, we check dependencies of sentences containing the word. We retain sentences in which all the dependencies and collocates are in the most frequent dependencies and collocates generated in section 4.2. These sentences are then returned as the representative sample of each word. The sample is then used to generate grammar patterns for each word. The representative sample of “避免” is shown in Table 3.

### 4.4 Converting to Grammar Patterns

In the fourth stage of the induction process, we convert dependencies into grammar patterns and collect pattern-associated collocations. For example, we convert argument type dependencies (e.g., nsubj, dobj) into the PoS of the dependant to represent the composition the grammar pattern. For prepositions, we use the preposition itself instead of the PoS to represent the composition the grammar pattern (e.g., “V 對 n”). The extraction process of grammar pat-

terns of “公佈” is shown in Figure 3.

### 4.5 Ranking Patterns and Collocations

In the final stage of the induction process, we group collocations with the same grammar pattern. Then we rank these patterns and related collocations by their frequencies. In this step, each pattern template will be extracted, and collocations and source sentences will be shown for users (the result of “公佈” is shown in Table 4).

This study set out to develop an interactive writing assistant which provides Chinese as second language (CSL) learners with pattern grammar (cf. (Hunston, 2000) for English pattern grammar) of a Chinese word, corresponding examples, and the frequency, helping CSL learners to use a word authentically in writing. Our system, *SmartWrite* employs an approach similar to *WriteAhead* (Yen et al., 2015) with a distinctive difference. As *WriteAhead* induces and displays English grammar, we assist users with common

| Representative sentences | dependency | Example sequences of processing a sentence   |
|--------------------------|------------|--|
| 但仍無法避免衝突                 | dobj       | 1. Sentence: 上海 世博會 日前 公佈 第二批 餐飲 服務 供應商 名單<br>2. Stanford Parser results:<br>[nn(世博會-2, 上海-1), nsubj(公佈-4, 世博會-2), tmod(公佈-4, 日前-3), root(ROOT-0, 公佈-4), ordmod(批-6, 第二-5), clf(名單-10, 批-6), nn(名單-10, 餐飲-7), nn(名單-10, 服務-8), nn(名單-10, 供應商-9), dobj(公佈-4, 名單-10)]<br>3. Ignore modifier <i>nsubj</i> in the head of <i>root</i> and <i>tmod</i> (modifiers).<br>4. Dependencies of headword “公佈”: nsubj:世博會, 公佈, dobj:名單<br>5. Collocation and pattern “公佈”: V n, “名單” |
| 避免突發意外                   | dobj       |  |
| 因此要避免此刻板印象之發生            | dobj       |  |
| 避免用藥困擾                   | dobj       |  |
| 則能有效避免風險產生               | comp       |  |
| 避免衝突                     | dobj       |  |
| 避免異物留在體內造成感染             | comp       |  |
| 就能避免精神官能症產生              | comp       |  |
| 還避免了人民幣匯率風險              | dobj       |  |
| 避免不必要的傳染風險               | dobj       |  |
| 才可以避免心血管疾病的發生            | dobj       |  |
| 而無法避免手術風險的發生             | dobj       |  |
| 因此要避免此刻板印象之發生            | dobj       |  |
| 這樣或許可以輕鬆避免爭議             | dobj       |  |

Table 3: Example sentences selected for the headword “公佈”.

| Pattern | Collocation |       |
|---------|-------------|-------|
| V n     | 公佈 結果       | 3,299 |
| p n V   | 在 公佈        | 868   |
| p n V n | 根據 公佈 數據    | 127   |
| V v     | 公佈 為        | 834   |
| V loc   | 公佈 後        | 380   |

Table 4: Example patterns extracted for the headword “公佈”

Chinese patterns and examples extracted from the UDN Corpus (as shown in Figure1). *SmartWrite* is available at <https://nlp-ultron.cs.nthu.edu.tw/smartwrite/>.

## 5 Experiments and Results

*SmartWrite* was designed to induce grammar patterns that are useful for assisted writing. As such, *SmartWrite* will be evaluated over a set verbs, which are central to writing sentences. Furthermore, since the goal is to retrieve good patterns, *SmartWrite* is evaluated mainly on the pattern level.

### 5.1 Experimental Setup

United Digital News (UDN) and United Evening News (UEN) are used for extracting grammar patterns and accompanying collocations. Both corpora consist of news articles from 2004 to 2017. There are approximately 2.3 million news articles with

Figure 3: Example of a segmented sentence to our format

over 679 million words.

To cope with common Chinese segmentation errors, we use Revised Mandarin Chinese Dictionary ([dict.revised.moe.edu.tw/cbdic/](http://dict.revised.moe.edu.tw/cbdic/)) to merge over-segmented tokens (e.g., “交流 (communicate)” and “道 (road)” will be merged into word “交流道 (interchange)”). We also removed 15% long sentences to reduce parsing errors. We ended up with 81,609,016 parsed sentences.

Subsequently, we parsed each sentence into dependencies using the Stanford Parser with a Chinese model (Chang et al., 2009). To speed up parsing time, we adopted the MapReduce framework to process all sentences in 5 days.

### 5.2 Results

Table 5 shows the average accuracy rates for patterns, collocations, and sentences over 16 verbs are around 70%. As we can see, the average accuracy rates various widely from 50% to 99%. That indicates that there is much room for improvement and a one-size-fit-all approach does not work very well. Also note that the accuracy rates of collocations and sentences tend to depend on that of patterns.

We take advantage of the widely available univer-

| Word | Pattern | Collocation | Sentence |
|------|---------|-------------|----------|
| 下跌   | .50     | .54         | .50      |
| 住    | .70     | .74         | .64      |
| 公佈   | .88     | .71         | .73      |
| 加入   | .60     | .56         | .49      |
| 放    | .70     | .78         | .68      |
| 穿    | .80     | .76         | .71      |
| 考慮   | .50     | .63         | .62      |
| 避免   | .80     | .89         | .84      |
| 邀請   | .50     | .61         | .49      |
| 交流   | 1.00    | .97         | .83      |
| 保護   | .75     | .82         | .85      |
| 作為   | .43     | .48         | .45      |
| 具有   | .50     | .74         | .73      |
| 努力   | .90     | .86         | .83      |
| 抵達   | .60     | .57         | .51      |
| 給予   | .90     | .76         | .61      |
| 維持   | .70     | .64         | .55      |
| 達成   | .60     | .63         | .51      |
| Avg. | .68     | .70         | .68      |

Table 5: Accuracy rate of generated patterns, collocations, and sentences

sal dependency parsers to induce lexical grammar patterns. Dependency supervision yields substantial improvements over unsupervised approaches, and the model is proven to be robust cross-linguistically, making it widely applicable for generating broad scale CCG treebanks for multiple languages.

## 6 Conclusions

Many avenues exist for future research and improvement of our system. For example, existing methods for prepositional phrase attachment could be implemented to avoid including adverbial prepositional phrases in the patterns. Additionally, an interesting direction to explore is to make the sample process more effective. Yet another direction of research would be to base the filtering process on supervised learning, so we can set the number of patterns retained to optimize the accuracy rate. On the other hand, the resulting patterns and collocations extracted from the large-scale corpus crucially reveals many interesting phenomena for linguistic researchers. For example, adverbs and other adjuncts are widely considered optional, but the data show

that some adverbs are obligatory when the object is omitted or fronted to the verb position (e.g., 塞車情況\*(難以)避免, 乳液\*(自己)買). The phenomenon uncovered by SmartWrite can be surveyed and explained by linguists and the resulting generalization or insight can also be incorporated into the system and improve the performance. In summary, we have described a method for inducing grammar patterns from a given sentence using an existing dependency parser. The method involves parsing sentences into dependencies, selecting representative sentences, converting dependencies into patterns, and ranking and filtering patterns. We have implemented and evaluated the method as applied to a very large-scale corpus. In a preliminary evaluations, we have shown that using active-learning allow us to cope with parsing errors and derive reasonably accurate patterns for assisted writing. This study lays the groundwork for future research into developing a computer assisted writing system for Chinese.

## References

- Vít Baisa and Vít Suchomel. 2014. Skell: Web interface for english language learning. In *Eighth Workshop on Recent Advances in Slavonic Natural Language Processing*, pages 63–70.
- Alex Boulton and Tom Cobb. 2017. Corpus use in language learning: A meta-analysis. *Language Learning*, 67(2):348–393.
- Pi-Chuan Chang, Huihsin Tseng, Daniel Jurafsky, and Christopher D. Manning. 2009. Discriminative reordering with chinese grammatical relations features. In *SSST@HLT-NAACL*.
- Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. Ltp: A chinese language technology platform. In *COLING*.
- Lynne Flowerdew. 2009. Applying corpus linguistics to pedagogy. *International Journal of Corpus Linguistics*, 14(3):393–417.
- Renfen Hu, Jiayong Chen, and Kuang hua Chen. 2016. The construction of a chinese collocational knowledge resource and its application for second language acquisition. In *COLING*.
- and Gill Francis. Hunston, Susan. 2000. *Pattern grammar: A corpus-driven approach to the lexical grammar of English*.
- Tim Johns. 1991. Should you be persuaded: Two samples of data-driven learning materials.

- Adam Kilgarriff, P. Rychly, P. Smrz, and D. Tugwell. 2004. The sketch engine. In *Proc Eleventh EURALEX International Congress*.
- Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. 2014. The sketch engine: ten years on. *Lexicography*, 1(1):7–36.
- Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *COLING*.
- P Rychly. 2008. A lexicographer-friendly association score. In *Proceedings of Recent Advances in Slavonic Natural Language Processing*.
- Shaoqun Wu and Ian Witten. 2016. Transcending concordance: Augmenting academic text for L2 writing. *International Journal of Computer-Assisted Language Learning and Teaching (IJCALLT)*, 6(2):1–18.
- Tzu-Hsi Yen, Jian-Cheng Wu, Jim Chang, Joanne Boisson, and Jason S. Chang. 2015. Writeahead: Mining grammar patterns in corpora for assisted writing. In *ACL*.