# Comparing the value of Latent Semantic Analysis on two English-to-Indonesian lexical mapping tasks

**Eliza Margaretha**
Faculty of Computer Science
University of Indonesia
Depok, Indonesia
elm40@ui.edu

**Ruli Manurung**
Faculty of Computer Science
University of Indonesia
Depok, Indonesia
maruli@cs.ui.ac.id

## Abstract

This paper describes an experiment that attempts to automatically map English words and concepts, derived from the Princeton WordNet, to their Indonesian analogues appearing in a widely-used Indonesian dictionary, using Latent Semantic Analysis (LSA). A bilingual semantic model is derived from an English-Indonesian parallel corpus. Given a particular word or concept, the semantic model is then used to identify its neighbours in a high-dimensional semantic space. Results from various experiments indicate that for bilingual word mapping, LSA is consistently outperformed by the basic vector space model, i.e. where the full-rank word-document matrix is applied. We speculate that this is due to the fact that the 'smoothing' effect LSA has on the word-document matrix, whilst very useful for revealing implicit semantic patterns, blurs the cooccurrence information that is necessary for establishing word translations.

## 1 Overview

An ongoing project at the Information Retrieval Lab, Faculty of Computer Science, University of Indonesia, concerns the development of an Indonesian WordNet[1]. To that end, one major task concerns the mapping of two monolingual dictionaries at two different levels: bilingual word mapping, which seeks to find translations of a lexical entry from one language to another, and bilingual concept mapping, which defines equivalence classes over concepts defined in two language resources. In other words, we try to automatically construct two variants of a bilingual dictionary between two languages, i.e. one with sense disambiguated entries and one without.

In this paper we present an extension to LSA into a bilingual context that is similar to (Rehder et al., 1997; Clodfelder, 2003; Deng and Gao, 2007), and then apply it to the two mapping tasks described above, specifically towards the lexical resources of Princeton WordNet (Fellbaum, 1998), an English semantic lexicon, and the *Kamus Besar Bahasa Indonesia* (KBBI)[2], considered by many to be the official dictionary of the Indonesian language.

We first provide formal definitions of our two tasks of bilingual word and concept mapping (Section 2) before discussing how these tasks can be automated using LSA (Section 3). We then present our experiment design and results (Sections 4 and 5) followed by an analysis and discussion of the results in Section 6.

## 2 Task Definitions

As mentioned above, our work concerns the mapping of two monolingual dictionaries. In our work, we refer to these resources as WordNets due to the fact that we view them as semantic lexicons that index entries based on meaning. However, we do not consider other semantic relations typically associated with a WordNet such as hypernymy, hyponymy, etc. For our purposes, a WordNet can be

[1] http://bahasa.cs.ui.ac.id/iwn

[2] The KBBI is the copyright of the Language Centre, Indonesian Ministry of National Education.

formally defined as a 4-tuple $(C, W, \chi, \omega)$ as follows:

- A concept $c \in C$ is a semantic entity, which represents a distinct, specific meaning. Each concept is associated with a gloss, which is a textual description of its meaning. For example, we could define two concepts, $c_1$ and $c_2$, where the former is associated with the gloss "*a financial institution that accepts deposits and channels the money into lending activities*" and the latter with "*sloping land (especially the slope beside a body of water*".
- A word $w \in W$ is an orthographic entity, which represents a word in a particular language (in the case of Princeton WordNet, English). For example, we could define two words, $w_1$ and $w_2$, where the former represents the orthographic string *bank* and the latter represents *spoon*.
- A word may convey several different concepts. The function $\chi: W \rightarrow P(C)$ returns all concepts conveyed by a particular word. Thus, $\chi(w)$, where $w \in W$, returns $C_w \subset C$, the set of all concepts that can be conveyed by w. Using the examples above, $\chi(w_1) = \{c_1, c_2\}$.
- Conversely, a concept may be conveyed by several words. The function $\omega: C \rightarrow P(w)$ returns all words that can convey a particular concept. Thus, $\omega(c)$, where $c \in C$, returns $W_c \subset W$, the set of all words that convey c. Using the examples above, $\omega(c_1) = \omega(c_2) = \{w_1\}$.

We can define different WordNets for different languages, e.g. $N^e = (C^e, T^e, \chi^e, \omega^e)$ and $N^i = (C^i, T^i, \chi^i, \omega^i)$. We also introduce the notation $w_i^x$ to denote word $i$ in $W^x$ and $c_j^x$ to denote concept $j$ in $C^x$. For the sake of our discussion, we will assume $N^e$ to be an English WordNet, and $N^i$ to be an Indonesian WordNet.

If we make the assumption that concepts are language independent, $N^e$ and $N^i$ should theoretically share the same set of universal concepts, $C$. In practice, however, we may have two WordNets with different conceptual representations, hence the distinction between $C^e$ and $C^i$. We introduce the relation $E: C^e \times C^i$ to denote the explicit mapping of equivalent concepts in $C^e$ and $C^i$.

We now describe two tasks that can be performed between $N^e$ and $N^i$, namely bilingual concept mapping and bilingual word mapping.

The task of bilingual concept mapping is essentially the establishment of the concept equivalence relation E. For example, given the example concepts in Table 1, bilingual concept mapping seeks to establish $E = \{(c_1^e, c_1^i), (c_1^e, c_2^i), (c_2^e, c_3^i), (c_3^e, c_4^e)\}$.

| Concept | Word | Gloss | Example |
|---|---|---|---|
| $c_1^e$ | $w_{time}^e$ | an instance or single occasion for some event | *"this **time** he succeeded"* |
| $c_2^e$ | $w_{time}^e$ | a suitable moment | *"it is **time** to go"* |
| $c_3^e$ | $w_{time}^e$ | a reading of a point in time as given by a clock (a word signifying the frequency of an event) | *"do you know what **time** it is?"* |
| $c_1^i$ | $w_{kali}^i$ | kata untuk menyatakan kekerapan tindakan (a word signifying a particular instance of an ongoing series of events) | *"dalam satu minggu ini, dia sudah empat **kali** datang ke rumahku" (this past week, she has come to my house four **times**)* |
| $c_2^i$ | $w_{kali}^i$ | kata untuk menyatakan salah satu waktu terjadinya peristiwa yg merupakan bagian dari rangkaian peristiwa yg pernah dan masih akan terus terjadi (a word signifying a particular instance of an ongoing series of events) | *"untuk **kali** ini ia kena batunya" (this **time** he suffered for his actions)* |
| $c_3^i$ | $w_{waktu}^i$ | saat yg tertentu untuk melakukan sesuatu (a specific time to be doing something) | *"**waktu** makan" (eating **time**)* |
| $c_4^i$ | $w_{jam}^i$ | saat tertentu, pada arloji jarumnya yg pendek menunjuk angka tertentu dan jarum panjang menunjuk angka 12 (the point in time when the short hand of a clock points to a certain hour and the long hand points to 12) | *"ia bangun **jam** lima pagi" (she woke up at five **o'clock**)* |
| $c_5^i$ | $w_{kali}^i$ | sebuah sungai yang kecil (a small river) | *"air di **kali** itu sangat keruh" (the water in that **small river** is very murky)* |

**Table 1. Sample Concepts in $C^e$ and $C^i$**

The task of bilingual word mapping is to find, given word $w_x^e \in W^e$, the set of all its plausible translations in $W^i$, regardless of the concepts being conveyed. We can also view this task as computing the union of the set of all words in $W^i$ that convey the set of all concepts conveyed by $w_x^e$. Formally, we compute the set $\{w_y^i : w_y^i \in \omega^i(c^i)$ where $(c^e, c^i) \in E$ and $c^e \in \chi^e(w_x^e)\}$.

For example, in Princeton WordNet, given $w_{time}^e$ (i.e. the English orthographic form time), $\chi^e(w_{time}^e)$ returns more than 15 different concepts, among others $\{c_1^e, c_2^e, c_3^e\}$ (see Table 1).

In Indonesian, assuming the relation $E$ as defined above, the set of words that convey $c_1^i$, i.e. $\omega^i(c_1^i)$, includes $w_{kali}^i$ (as in "kali ini dia berhasil" = "this time she succeeded").

On the other hand, $\omega^i(c_3^i)$ may include $w_{waktu}^i$ (as in "ini waktunya untuk pergi" = "it is time to go") and $w_{saat}^i$ (as in "sekarang saatnya menjual saham" = "now is the time to sell shares"), and lastly, $\omega^i(c_4^i)$ may include $w_{jam}^i$ (as in "apa anda tahu jam berapa sekarang?" = "do you know what time it is now?").

Thus, the bilingual word mapping task seeks to compute, for the English word $w_{time}^e$, the set of Indonesian words $\{w_{kali}^i, w_{waktu}^i, w_{saat}^i, w_{jam}^i, \dots\}$. Note that each of these Indonesian words may convey different concepts, e.g. $\chi^i(w_{kali}^i)$ may include $c_5^i$ in Table 1.

## 3 Automatic mapping using Latent Semantic Analysis

Latent semantic analysis, or simply LSA, is a method to discern underlying semantic information from a given corpus of text, and to subsequently represent the contextual meaning of the words in the corpus as a vector in a high-dimensional semantic space (Landauer et al., 1998). As such, LSA is a powerful method for word sense disambiguation.

The mathematical foundation of LSA is provided by the Singular Value Decomposition, or SVD. Initially, a corpus is represented as an $n \times m$ word-passage matrix $M$, where cell $[n, m]$ represents the occurrence of the $n$-th word in the $m$-th passage. Thus, each row of $M$ represents a word and each column represents a passage. The SVD is then applied to $M$, decomposing it such that $M = USV^T$, where $U$ is an $m \times m$ matrix of left singular vectors, $V^T$ is an $n \times n$ matrix of right singular vectors, and $\Sigma$ is an $n \times m$ matrix containing the singular values of $M$.

Crucially, this decomposition factors $M$ using an orthonormal basis that produces an optimal reduced rank approximation matrix (Kalman, 1996). By reducing dimensions of the matrix irrelevant information and noise are removed. The optimal rank reduction yields useful induction of implicit relations. However, finding the optimal level of rank reduction is an empirical issue.

LSA can be applied to exploit a parallel corpus to automatically perform bilingual word and concept mapping. We define a parallel corpus $P$ as a set of pairs $p = (d_e, d_i)$, where $d_e$ is a document written in the language of $N^e$, and $d_i$ is its translation in the language of $N^i$.

Intuitively, we would expect that if two words $w_x^e$ and $w_x^i$ consistently occur in documents that are translations of each other, but not in other documents, that they would at the very least be semantically related, and possibly even be translations of each other. For instance, imagine a parallel corpus consisting of news articles written in English and Indonesian: in English articles where the word *Japan* occurs, we would expect the word *Jepang* to occur in the corresponding Indonesian articles.

This intuition can be represented in a word-document matrix as follows: let $M_E$ be a word-document matrix of $m$ English documents and $n_E$ English words, and $M_I$ be a word-document matrix of $m$ Indonesian documents and $n_I$ Indonesian words. The documents are arranged such that, for $1 \leq j \leq m$, the English document represented by column $j$ of $M_E$ and the Indonesian document represented by column $j$ of $M_I$ form a pair of translations. Since they are translations, we can view them as occupying exactly the same point in semantic space, and could just as easily view column $j$ of both matrices as representing the union, or concatenation, of the two articles.

Consequently, we can construct the bilingual word-document matrix

$$M = \begin{bmatrix} M_E \\ M_I \end{bmatrix}$$

which is an $(n_E + n_I) \times m$ matrix where cell $[i, j]$ contains the number of occurrences of word $i$ in article $j$. Row $i$ forms the semantic vector of, for $i \leq n_E$, an English word, and for $i > n_E$, an Indo-

nesian word. Conversely, column $j$ forms a vector representing the English and Indonesian words appearing in translations of document $j$.

This approach is similar to that of (Rehder et al., 1997; Clodfelder, 2003; Deng and Gao, 2007). The SVD process is the same, while the usage is different. For example, (Rehder et al., 1997) employ SVD for cross language information retrieval. On the other hand, we use it to accomplish word and concept mappings.

LSA can be applied to this bilingual word-document matrix. Computing the SVD of this matrix and reducing the rank should unearth implicit patterns of semantic concepts. The vectors representing English and Indonesian words that are closely related should have high similarity; word translations more so.

To approximate the bilingual word mapping task, we compare the similarity between the semantic vectors representing words in $W^e$ and $W^i$. Specifically, for the first $n_E$ rows in $M$ which represent words in $W^e$, we compute their similarity to each of the last $n_I$ rows which represent words in $W^i$. Given a large enough corpus, we would expect all words in $W^e$ and $W^i$ to be represented by rows in $M$.

To approximate the bilingual concept mapping task, we compare the similarity between the semantic vectors representing concepts in $C^e$ and $C^i$. These vectors can be approximated by first constructing a set of textual context representing a concept $c$. For example, we can include the words in $\omega(c)$ together with the words from its gloss and example sentences. The semantic vector of a concept is then a weighted average of the semantic vectors of the words contained within this context set, i.e. rows in $M$. Again, given a large enough corpus, we would expect enough of these context words to be represented by rows in M to form an adequate semantic vector for the concept $c$.

# 4 Experiments

## 4.1 Existing Resources

For the English lexicon, we used the most current version of WordNet (Fellbaum, 1998), version 3.0[3]. For each of the 117659 distinct synsets, we only use the following data: the set of words belonging to the synset, the gloss, and example sentences, if any. The union of these resources yields a set 169583 unique words.

For the Indonesian lexicon, we used an electronic version of the KBBI developed at the University of Indonesia. For each of the 85521 distinct word sense definitions, we use the following data: the list of sublemmas, i.e. inflected forms, along with gloss and example sentences, if any. The union of these resources yields a set of 87171 unique words.

Our main parallel corpus consists of 3273 English and Indonesian article pairs taken from the ANTARA news agency. This collection was developed by Mirna Adriani and Monica Lestari Paramita at the Information Retrieval Lab, University of Indonesia[4].

A bilingual English-Indonesia dictionary was constructed using various online resources, including a handcrafted dictionary by Hantarto Widjaja[5], kamus.net, and Transtool v6.1, a commercial translation system. In total, this dictionary maps 37678 unique English words to 60564 unique Indonesian words.

## 4.2 Bilingual Word Mapping

Our experiment with bilingual word mapping was set up as follows: firstly, we define a collection of article pairs derived from the ANTARA collection, and from it we set up a bilingual word-document matrix (see Section 3). The LSA process is subsequently applied on this matrix, i.e. we first compute the SVD of this matrix, and then use it to compute the optimal $k$-rank approximation. Finally, based on this approximation, for a randomly chosen set of vectors representing English words, we compute the $n$ nearest vectors representing the $n$ most similar Indonesian words. This is conventionally computed using the cosine of the angle between two vectors.

Within this general framework, there are several variables that we experiment with, as follows:

- **Collection size**. Three subsets of the parallel corpus were randomly created: $P_{100}$ contains 100 article pairs, $P_{500}$ contains 500 article pairs, and $P_{1000}$ contains 1000 article pairs. Each subsequent subset wholly contains the previous subsets, i.e. $P_{100} \subset P_{500} \subset P_{1000}$.

[3] More specifically, the SQL version available from http://wnsqlbuilder.sourceforge.net

[4] publication forthcoming

[5] http://hantarto.definitionroadsafety.org

- **Rank reduction**. For each collection, we applied LSA with different degrees of rank approximation, namely 10%, 25%, and 50% the number of dimensions of the original collection. Thus, for $P_{100}$ we compute the 10, 25, and 50-rank approximations, for $P_{500}$ we compute the 50, 125, and 250-rank approximations, and for $P_{1000}$ we compute the 100, 250, and 500-rank approximations.
- **Removal of stopwords**. Stopwords are words that appear numerously in a text, thus are assumed as insignificant to represent the specific context of the text. It is a common technique used to improve performance of information retrieval systems. It is applied in preprocessing the collections, i.e. removing all instances of the stopwords in the collections before applying LSA.
- **Weighting**. Two weighting schemes, namely TF-IDF and Log-Entropy, were applied to a word-document matrix separately.
- **Mapping Selection**. For computing the precision and recall values, we experimented with the number of mapping results to consider: the top 1, 10, 50, and 100 mappings based on similarity were taken.

| | |
|---|---|
| film | 0.814 |
| filmnya | 0.698 |
| sutradara | 0.684 |
| garapan | 0.581 |
| perfilman | 0.554 |
| penayangan | 0.544 |
| kontroversial | 0.526 |
| koboi | 0.482 |
| irasional | 0.482 |
| frase | 0.482 |

(a)

| | |
|---|---|
| pembebanan | 0.973 |
| kijang | 0.973 |
| halmahera | 0.973 |
| alumina | 0.973 |
| terjadwal | 0.973 |
| viskositas | 0.973 |
| tabel | 0.973 |
| royalti | 0.973 |
| reklamasi | 0.973 |
| penyimpan | 0.973 |

(b)

**Table 2. The Most 10 Similar Indonesian Words for the English Words (a) Film and (b) Billion**

As an example, Table 2 presents the results of mapping $w^e_{film}$ and $w^e_{billion}$, i.e. the two English words *film* and *billion*, respectively to their Indonesian translations, using the $P_{1000}$ training collection with 500-rank approximation. No weighting was applied. The former shows a successful mapping, while the latter shows an unsuccessful one. Bilingual LSA correctly maps $w^e_{film}$ to its translation, $w^i_{film}$, despite the fact that they are treated as separate elements, i.e. their shared orthography is completely coincidental. Additionally, the other Indonesian words it suggests are semantically related, e.g. *sutradara* (director), *garapan* (creation), *penayangan* (screening), etc. On the other hand, the suggested word mappings for $w^e_{billion}$ are incorrect, and the correct translation, *milyar*, is missing. We suspect this may be due to several factors. Firstly, *billion* does not by itself invoke a particular semantic frame, and thus its semantic vector might not suggest a specific conceptual domain. Secondly, *billion* can sometimes be translated numerically instead of lexically. Lastly, this failure may also be due to the lack of data: the collection is simply too small to provide useful statistics that represent semantic context. Similar LSA approaches are commonly trained on collections of text numbering in the tens of thousands of articles.

Note as well that the absolute vector cosine values do not accurately reflect the correctness of the word translations. To properly assess the results of this experiment, evaluation against a gold standard is necessary. This is achieved by comparing its precision and recall against the Indonesian words returned by the bilingual dictionary, i.e. how isomorphic is the set of LSA-derived word mappings with a human-authored set of word mappings?

We provide a baseline as comparison, which computes the nearness between English and Indonesian words on the original word-document occurrence frequency matrix. Other approaches are possible, e.g. mutual information (Sari, 2007).

Table 3(a)-(e) shows the different aspects of our experiment results by averaging the other variables. Table 3(a) confirms our intuition that as the collection size increases, the precision and recall values also increase. Table 3(b) presents the effects of rank approximation. It shows that the higher the rank approximation percentage, the better the mapping results. Note that a rank approximation of 100% is equal to the FREQ baseline of simply using the full-rank word-document matrix for computing vector space nearness. Table 3(c) suggests that stopwords seem to help LSA to yield the correct mappings. It is believed that stopwords are not bounded by semantic domains, thus do not carry any semantic bias. However, on account of the small size of the collection, in coincidence, stopwords, which consistently appear in a specific domain, may carry some semantic information about the domain. Table 3(d) compares the mapping results in terms of weighting usage. It sug-

gests that weighting can improve the mappings. Additionally, Log-Entropy weighting yields the highest results. Table 3(e) shows the comparison of mapping selections. As the number of translation pairs selected increases, the precision value decreases. On the other hand, as the number of translation pairs selected increases, the possibility to find more pairs matching the pairs in bilingual dictionary increases. Thus, the recall value increases as well.

| Collection Size | FREQ | | LSA | |
|---|---|---|---|---|
| | P | R | P | R |
| $P_{100}$ | 0.0668 | 0.1840 | 0.0346 | 0.1053 |
| $P_{500}$ | 0.1301 | 0.2761 | 0.0974 | 0.2368 |
| $P_{1000}$ | **0.1467** | **0.2857** | 0.1172 | 0.2603 |

(a)

| Rank Approximation | P | R |
|---|---|---|
| **10%** | 0.0680 | 0.1727 |
| **25%** | 0.0845 | 0.2070 |
| **50%** | 0.0967 | 0.2226 |
| **100%** | **0.1009** | **0.2285** |

(b)

| Stopwords | FREQ | | LSA | |
|---|---|---|---|---|
| | P | R | P | R |
| **Contained** | 0.1108 | **0.2465** | 0.0840 | 0.2051 |
| **Removed** | **0.1138** | 0.2440 | 0.0822 | 0.1964 |

(c)

| Weighting Usage | FREQ | | LSA | |
|---|---|---|---|---|
| | P | R | P | R |
| **No Weighting** | 0.1009 | 0.2285 | 0.0757 | 0.1948 |
| **Log-Entropy** | **0.1347** | **0.2753** | 0.1041 | 0.2274 |
| **TF-IDF** | 0.1013 | 0.2319 | 0.0694 | 0.1802 |

(d)

| Mapping Selection | FREQ | | LSA | |
|---|---|---|---|---|
| | P | R | P | R |
| **Top 1** | **0.3758** | 0.1588 | 0.2380 | 0.0987 |
| **Top 10** | 0.0567 | 0.2263 | 0.0434 | 0.1733 |
| **Top 50** | 0.0163 | 0.2911 | 0.0133 | 0.2338 |
| **Top 100** | 0.0094 | **0.3183** | 0.0081 | 0.2732 |

(e)

**Table 3. Results of bilingual word mapping comparing (a) collection size, (b) rank approximation, (c) removal of stopwords, (d) weighting schemes, and (e) mapping selection**

Most interestingly, however, is the fact that the FREQ baseline, which uses the basic vector space model, consistently outperforms LSA.

### 4.3 Bilingual Concept Mapping

Using the same resources from the previous experiment, we ran an experiment to perform bilingual concept mapping by replacing the vectors to be compared with semantic vectors for concepts (see Section 3). For concept $c^e \in C^e$, i.e. a WordNet synset, we constructed a set of textual context as the union of $\omega(c)$, the set of words in the gloss of $c^e$, and the set of words in the example sentences associated with $c^e$. To represent our intuition that the words in $\omega(c)$ played more of an important role in defining the semantic vector than the words in the gloss and example, we applied a weight of 60%, 30%, and 10% to the three components, respectively. Similarly, a semantic vector representing a concept $c^i \in C^i$, i.e. an Indonesian word sense in the KBBI, was constructed from a textual context set composed of the sublemma, the definition, and the example of the word sense, using the same weightings. We only average word vectors if they appear in the collection (depending on the experimental variables used).

We formulated an experiment which closely resembles the word sense disambiguation problem: given a WordNet synset, the task is to select the most appropriate Indonesian sense from a subset of senses that have been selected based on their words appearing in our bilingual dictionary. These specific senses are called suggestions. Thus, instead of comparing the vector representing *communication* with every single Indonesian sense in the KBBI, in this task we only compare it against suggestions with a limited range of sublemmas, e.g. *komunikasi*, *perhubungan*, *hubungan*, etc.

This setup is thus identical to that of an ongoing experiment here to manually map WordNet synsets to KBBI senses. Consequently, this facilitates assessment of the results by computing the level of agreement between the LSA-based mappings with human annotations.

To illustrate, Table 4(a) and 4(b) presents a successful and unsuccessful example of mapping a WordNet synset. For each example we show the synset ID and the ideal textual context set, i.e. the set of words that convey the synset, its gloss and example sentences. We then show the actual textual context set with the notation {{X}, {Y}, {Z}}, where X, Y , and Z are the subset of words that appear in the training collection. We then show the Indonesian word sense deemed to be most similar. For each sense we show the vector similarity score, the KBBI ID and its ideal textual context set, i.e. the sublemma, its definition and example sen-

tences. We then show the actual textual context set with the same notation as above.

| |
|---|
| **WordNet Synset ID**: 100319939, **Words**: chase, following, pursual, pursuit, **Gloss**: the act of pursuing in an effort to overtake or capture, **Example**: the culprit started to run and the cop took off in pursuit, **Textual context set**: {{following, chase}, {the, effort, of, to, or, capture, in, act, pursuing, an}, {the, off, took, to, run, in, culprit, started, and}} |
| **KBBI ID**: k39607 - **Similarity**: 0.804, **Sublemma**: mengejar, **Definition**: berlari untuk menyusul menangkap dsb memburu, **Example**: ia berusaha mengejar dan menangkap saya, **Textual context set**: {{mengejar}, {memburu, berlari, menangkap, untuk, menyusul},{berusaha, dan, ia, mengejar, saya, menangkap}} |

(a)

| |
|---|
| **WordNet synset ID**: 201277784, **Words**: crease, furrow, wrinkle<br>**Gloss**: make wrinkled or creased, **Example**: furrow one's brow,<br>**Textual context set**: {{}, {or, make}, {s, one}} |
| **KBBI ID**: k02421 - **Similarity**: 0.69, **Sublemma**: alur, **Definition**: jalinan peristiwa dl karya sastra untuk mencapai efek tertentu pautannya dapat diwujudkan oleh hubungan temporal atau waktu dan oleh hubungan kausal atau sebab-akibat, **Example**: (none), **Textual context set**: {{alur}, {oleh, dan, atau, jalinan, peristiwa, diwujudkan, efek, dapat, karya, hubungan, waktu, mencapai, untuk, tertentu}, {}} |

(b)

**Table 4. Example of (a) Successful and (b) Unsuccessful Concept Mappings**

In the first example, the textual context sets from both the WordNet synset and the KBBI senses are fairly large, and provide sufficient context for LSA to choose the correct KBBI sense. However, in the second example, the textual context set for the synset is very small, due to the words not appearing in the training collection. Furthermore, it does not contain any of the words that truly convey the concept. As a result, LSA is unable to identify the correct KBBI sense.

For this experiment, we used the $P_{1000}$ training collection. The results are presented in Table 5. As a baseline, we select three random suggested Indonesian word senses as a mapping for an English word sense. The reported random baseline in Table 5 is an average of 10 separate runs. Another baseline was computed by comparing English common-based concepts to their suggestion based on a full rank word-document matrix. Top 3 Indonesian concepts with the highest similarity values are designated as the mapping results. Subsequently, we compute the Fleiss kappa (Fleiss, 1971) of this result together with the human judgements.

The average level of agreement between the LSA mappings 10% and the human judges (0.2713) is not as high as between the human judges themselves (0.4831). Nevertheless, in general it is better than the random baseline (0.2380) and frequency baseline (0.2132), which suggests that LSA is indeed managing to capture some measure of bilingual semantic information implicit within the parallel corpus.

Furthermore, LSA mappings with 10% rank approximation yields higher levels of agreement than LSA with other rank approximations. It is contradictory with the word mapping results where LSA with bigger rank approximations yields higher results (Section 4.2).

## 5 Discussion

Previous works have shown LSA to contribute positive gains to similar tasks such as Cross Language Information Retrieval (Rehder et al., 1997). However, the bilingual word mapping results presented in Section 4.3 show the basic vector space model consistently outperforming LSA at that particular task, despite our initial intuition that LSA should actually improve precision and recall.

We speculate that the task of bilingual word mapping may be even harder for LSA than that of

| Judges | Synsets | Fleiss Kappa Values | | | | | |
|---|---|---|---|---|---|---|---|
| | | Judges only | Judges + RNDM3 | Judges + FREQ Top 3 | Judges + LSA 10% Top3 | Judges + LSA 25% Top3 | Judges + LSA 50% Top3 |
| ≥ 2 | 144 | 0.4269 | 0.1318 | 0.1667 | 0.1544 | 0.1606 | 0.1620 |
| ≥ 3 | 24 | 0.4651 | 0.2197 | 0.2282 | 0.2334 | 0.2239 | 0.2185 |
| ≥ 4 | 8 | 0.5765 | 0.3103 | 0.2282 | 0.3615 | 0.3329 | 0.3329 |
| ≥ 5 | 4 | 0.4639 | 0.2900 | 0.2297 | 0.3359 | 0.3359 | 0.3359 |
| **Average** | | 0.4831 | 0.2380 | 0.2132 | 0.2713 | 0.2633 | 0.2623 |

**Table 5. Results of Concept Mapping**

bilingual concept mapping due to its finer alignment granularity. While concept mapping attempts to map a concept conveyed by a group of semantically related words, word mapping attempts to map a word with a specific meaning to its translation in another language.

In theory, LSA employs rank reduction to remove noise and to reveal underlying information contained in a corpus. LSA has a 'smoothing' effect on the matrix, which is useful to discover general patterns, e.g. clustering documents by semantic domain. Our experiment results, however, generally shows the frequency baseline, which employs the full rank word-document matrix, outperforming LSA.

We speculate that the rank reduction perhaps blurs some crucial details necessary for word mapping. The frequency baseline seems to encode more cooccurrence than LSA. It compares word vectors between English and Indonesian that contain pure frequency of word occurrence in each document. On the other hand, LSA encodes more semantic relatedness. It compares English and Indonesian word vectors containing estimates of word frequency in documents according to the context meaning. Since the purpose of bilingual word mapping is to obtain proper translations for an English word, it may be better explained as an issue of cooccurrence rather than semantic relatedness. That is, the higher the rate of cooccurrence between an English and an Indonesian word, the likelier they are to be translations of each other.

LSA may yield better results in the case of finding words with similar semantic domains. Thus, the LSA mapping results should be better assessed using a resource listing semantically related terms, rather than using a bilingual dictionary listing translation pairs. A bilingual dictionary demands more specific constraints than semantic relatedness, as it specifies that the mapping results should be the translations of an English word.

Furthermore, polysemous terms may become another problem for LSA. By rank approximation, LSA estimates the occurrence frequency of a word in a particular document. Since polysemy of English terms and Indonesian terms can be quite different, the estimations for words which are mutual translations can be different. For instance, *kali* and *waktu* are Indonesian translations for the English word *time*. However, *kali* is also the Indonesian translation for the English word *river*. Suppose *kali* and *time* appear frequently in documents about multiplication, but *kali* and *river* appear rarely in documents about river. Then, *waktu* and *time* appear frequently in documents about time. As a result, LSA may estimate *kali* with greater frequency in documents about multiplication and time, but with lower frequency in documents about river. The word vectors between *kali* and *river* may not be similar. Thus, in bilingual word mapping, LSA may not suggest *kali* as the proper translation for *river*. Although polysemous words can also be a problem for the frequency baseline, it merely uses raw word frequency vectors, the problem does not affect other word vectors. LSA, on the other hand, exacerbates this problem by taking it into account in estimating other word frequencies.

## 6 Summary

We have presented a model of computing bilingual word and concept mappings between two semantic lexicons, in our case Princeton WordNet and the KBBI, using an extension to LSA that exploits implicit semantic information contained within a parallel corpus.

The results, whilst far from conclusive, indicate that that for bilingual word mapping, LSA is consistently outperformed by the basic vector space model, i.e. where the full-rank word-document matrix is applied, whereas for bilingual concept mapping LSA seems to slightly improve results. We speculate that this is due to the fact that LSA, whilst very useful for revealing implicit semantic patterns, blurs the cooccurrence information that is necessary for establishing word translations.

We suggest that, particularly for bilingual word mapping, a finer granularity of alignment, e.g. at the sentential level, may increase accuracy (Deng and Gao, 2007).

## Acknowledgment

The work presented in this paper is supported by an RUUI (*Riset Unggulan Universitas Indonesia*) 2007 research grant from DRPM UI (*Direktorat Riset dan Pengabdian Masyarakat Universitas Indonesia*). We would also like to thank Franky for help in software implementation and Desmond Darma Putra for help in computing the Fleiss kappa values in Section 4.3.

## References


Eneko Agirre and Philip Edmonds, editors. 2007. *Word Sense Disambiguation: Algorithms and Applications.* Springer.

Katri A. Clodfelder. 2003. *An LSA Implementation Against Parallel Texts in French and English.* In Proceedings of the HLT-NAACL Workshop: Building and Using Parallel Texts: Data Driven Machine Translation and Beyond, 111–114.

Yonggang Deng and Yuqing Gao. June 2007. *Guiding statistical word alignment models with prior knowledge.* In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 1–8, Prague, Czech Republic. Association for Computational Linguistics.

Christiane Fellbaum, editor. May 1998. *WordNet: An Electronic Lexical Database.* MIT Press.

Joseph L. Fleiss. 1971.*Measuring nominal scale agreement among many raters.* Psychological Bulletin, 76(5):378–382.

Dan Kalman. 1996. *A singularly valuable decomposition: The svd of a matrix.* The College Mathematics Journal, 27(1):2–23.

Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. 1998. *An introduction to latent semantic analysis.* Discourse Processes, 25:259–284.

Bob Rehder, Michael L. Littman, Susan T. Dumais, and Thomas K. Landauer. 1997. *Automatic 3-language cross-language information retrieval with latent semantic indexing.* In Proceedings of the Sixth Text Retrieval Conference (TREC-6), pages 233–239.

Syandra Sari. 2007. *Perolehan informasi lintas bahasa indonesia-inggris berdasarkan korpus paralel dengan menggunakan metoda mutual information dan metoda similarity thesaurus.* Master's thesis, Faculty of Computer Science, University of Indonesia, Call number: T-0617.