MRD-based Word Sense Disambiguation: Extensions and Applications

Timothy Baldwin

THE UNIVERSITY OF MELBOURNE

Joint Work with F. Bond, S. Fujita, T. Tanaka, Willy and S.N. Kim

Research Objectives

- To develop/advance (unsupervised/supervised) baseline methods for MRD-based WSD (esp. for the Hinoki Sensebank)
- To apply/extend dictionary definition-based WSD methods over Japanese data
- To explore the use of sense-disambiguated ontologies in definition-based WSD

Japanese Semantic Lexicon (Lexeed)

- The most familiar words of Japanese
  - ⋆ Familiarity is estimated by psychological experiments
  - ⋆ All words with a familiarity ≥ 5.0 $(1 \ldots 7)$
- 28,000 words and 47,000 senses
- Covers 75% of tokens in a typical newspaper
- Rewritten so definition sentences use only basic words (and function words)

  $\rightarrow$ closed-world lexicon

Lexeed Dictionary – Dog_1

INDEX	犬 *inu*		
POS	noun		
FAMILIARITY	6.53 [1–7] FREQUENCY 67		
SENSE 1	DEFINITION	犬 科 の 食肉 動物 。	
		A carnivorous animal of the canidae family .	
		家畜 として 古く から 飼わ れ 、 飼い主 に 忠実 。	
		Kept domestically from ancient times; loyal to their owners.	
	EXAMPLE	犬 を 飼っ て いる 家 が 多い 。	
		There are many households that keep dogs.	

Hinoki Sensebank

- All content words in definition and example sentences 5-way sense annotated (and resolved to majority sense)
- All definitions and example sentences treebanked using JACY
- Ontology induced from sensebank based on parsed definition sentences, with relation types hyper/hyponyms, domains, meronyms and synonyms
- Senses also linked to GoiTaikei and WordNet

Hinoki Sensebank – Dog$_1$

INDEX	犬 *inu*		
POS	noun LEXICAL-TYPE `noun-lex`		
FAMILIARITY	6.53 [1–7] FREQUENCY 67 ENTROPY 0.03		
SENSE 1 0.99	DEFINITION	犬$_1$ 科 の 食肉$_1$ 動物$_1$ 。 A carnivorous animal of the canidae family . 家畜$_1$ として 古く$_1$ から 飼わ$_1$ れ 、 飼い主$_1$ に 忠実$_1$ 。 Kept domestically from ancient times; loyal to their owners.	
	EXAMPLE	犬$_1$ を 飼っ$_1$ て いる 家$_3$ が 多い$_1$ 。 There are many households that keep dogs.	
	HYPERNYM	動物$_1$ *dōbutsu* "animal"	
	SEM. CLASS	⟨`537:beast`⟩ (⊂ ⟨`535:animal`⟩)	
	WORDNET	*dog*$_1$	

Lexeed Dictionary – Dog$_2$

INDEX	犬 *inu*		
POS	noun		
FAMILIARITY	6.53 [1–7]		
SENSE 2	DEFINITION	警察 など の 回し者 。 スパイ 。 A secret agent for the police, etc. A spy.	
	EXAMPLE	警察 の 犬$_2$ だけ に は 成り たく ない 。 I want to turn into anything but a police spy.	

Hinoki Sensebank – Dog$_2$

INDEX	犬 *inu*			
POS	noun	LEXICAL-TYPE `noun-lex`		
FAMILIARITY	6.53 [1–7]	FREQUENCY 67	ENTROPY 0.03	
SENSE 2 0.01	DEFINITION	警察$_1$ など の 回し者$_1$ 。 スパイ$_1$ 。 A secret agent for the police, etc. A spy.		
	EXAMPLE	警察$_1$ の 犬$_2$ だけ に は 成り$_4$ たく ない 。 I want to turn into anything but a police spy.		
	HYPERNYM	回し者$_1$ *mawashimono* "secret agent"		
	SYNONYM	スパイ$_1$ *supai* "spy"		
	SEM. CLASS	⟨317:spy⟩ (⊂ ⟨317:spy⟩)		
	WORDNET	*spy*$_1$		

OK, OK ... but why MRD-based WSD?

- MRD-based WSD shown to provide very high unsupervised baseline (e.g. Lesk algorithm in Senseval tasks)
- Suitable for all words WSD tasks (no data bottleneck)
- MRDs have (relatively) high availability compared to sensebanked data
- MRD-based WSD is easily adaptable to new MRDs, languages

Basic Algorithm (Lesk++)

for each word w_i in context $\mathbf{w} = w_1 w_2 ... w_n$ **do**
  for each sense $s_{i,j}$ and definition $\mathbf{d}_{i,j}$ of w_i **do**
    $score(s_{i,j}) = similarity(\mathbf{w} \setminus w_i, \mathbf{d}_{i,j})$
  end for
  $s_i^* = \arg\max_j score(s_{i,j})$
end for

Original Lesk (Lesk, 1986)

- $similarity$ = simple set intersection

- Example:

  ⋆ Input: おとなしい $\boxed{\text{犬}}$ を 飼い たい → { おとなしい,飼う }
  ⋆ dog$_1$: { 犬,食肉,動物,家畜,古い,飼う,飼い主,忠実 }

$$similarity(\mathrm{Input}, \mathrm{dog}_1) = 1$$

  ⋆ dog$_2$: { 警察,回し者,スパイ }

$$similarity(\mathrm{Input}, \mathrm{dog}_2) = 0$$

Extended Lesk (Banerjee and Pedersen, 2003)

- $similarity$ defined relative to each context word w_j and RELPAIRS, e.g. $\{\langle def, def \rangle, \langle hype, hype \rangle, \langle hypo, hypo \rangle\}$:

$$similarity(w_j, w_i) = \sum_{\langle R_m, R_n \rangle \in \mathrm{RELPAIRS}} score(R_m(w_i), R_n(w_j))$$

- $score$ based on square of length of longest substring match

- Only ever compare definitions to definitions (never directly to context)

- Example:
  - ⋆ Input: おとなしい $\boxed{\text{犬}}$ を 飼い たい $\rightarrow$ { おとなしい,飼う }
  - ⋆ dog_1:

$$score(def(nice_1), def(dog_1)) + score(hype(nice_1), hype(dog_1)) +$$
$$score(hypo(nice_1), hypo(dog_1)) + score(def(nice_2), def(dog_1)) +$$
$$score(def(keep_1), def(dog_1)) + score(hype(keep_1), hype(dog_1)) +$$
$$score(def(keep_2), def(dog_1)) + \ldots$$

  - ⋆ dog_2: { 警察,回し者,スパイ }

$$score(def(nice_1), def(dog_2)) + score(hype(nice_1), hype(dog_2)) +$$
$$score(hypo(nice_1), hypo(dog_2)) + score(def(nice_2), def(dog_2)) +$$
$$score(def(keep_1), def(dog_2)) + score(hype(keep_1), hype(dog_2)) +$$
$$score(def(keep_2), def(dog_2)) + \ldots$$

Our Method

- *similarity* defined based on Dice coefficient:

$$sim_{\mathrm{DICE}}(A, B) = \frac{2\,|A \cap B|}{|A| + |B|}$$

- Similarly to basic Lesk, compare context words with definitions:
  - ⋆ Input: おとなしい $\boxed{犬}$ を 飼い たい → { おとなしい,飼う }
  - ⋆ dog_1: { 犬,食肉,動物,家畜,古い,飼う,飼い主,忠実 }

$$similarity(\mathrm{Input}, \mathrm{dog}_1) = \frac{2}{10}$$

  - ⋆ dog_2: { 警察,回し者,スパイ }

$$similarity(\mathrm{Input}, \mathrm{dog}_2) = \frac{0}{5}$$

- Similarly to Banerjee and Pedersen, 2003, expand definitions based on ontological relations, but into single expanded term vector (c.f. query expansion)

  ⋆ dog_1+hype: { 犬,食肉,動物,家畜,古い,飼う,飼い主,忠実,生物,大きな,区分 }

$$similarity(\text{Input}, \text{dog}_1) = \frac{2}{13}$$

  ⋆ dog_2+hype: { 警察,回し者,スパイ }

$$similarity(\text{Input}, \text{dog}_2) = \frac{0}{5}$$

- Different to Banerjee and Pedersen, 2003, expand out the definition to include the definition of each content word (context-sensitive or context-insensitive)

- Example (sense-sensitive):

  ⋆ dog$_1$+*hype*: { 犬,犬,食肉,...,食肉,猛獣,他,...,動物,,生物,大きな,.......,忠実 }

  ⋮

- Example (sense-insensitive):

  ⋆ dog$_1$+*hype*: { 犬,犬,食肉,...,警察,回し者,スパイ,食肉,猛獣,他,...,食用,する,...,動物,...,忠実 }

  ⋮

The Nitty-gritty Details

- Ontological relations: hypernymy, hyponymy and synonymy (only)
- Token representation: characters vs. words
- Evaluate in terms of simple accuracy (100%-recall method)

Outline of the Datasets

- Training data: Hinoki definition sentences
- Test datasets:
  - ⋆ Hinoki example sentences
  - ⋆ Senseval-2 Japanese dictionary task
- For all three datasets, each open-class word is multiply sense-annotated, and sense-arbitrated relative to the majority annotation

Results over the Hinoki Example Sentences

	Sense-sensitive		Sense-insensitive	
	Word	Char	Word	Char
unsupervised (random) baseline:	0.527			
supervised (first-sense) baseline:	0.633			
Banerjee and Pedersen, 2003	0.648			
simple	0.469	0.524	0.469	0.524
+syn	0.560	0.538	0.548	0.543
+hyper	0.559	0.539	0.548	0.537
+hypo	0.656	0.644	**0.655**	**0.644**
+hyper +hypo	0.648	0.641	0.629	0.630
+syn +hyper +hypo	0.650	0.633	0.627	0.623
+extdef	0.489	0.527	0.489	0.527
+extdef +syn	0.577	0.560	0.551	0.543
+extdef +hyper	0.577	0.563	0.551	0.542
+extdef +hypo	0.653	0.646	0.649	**0.644**
+extdef +hyper +hypo	**0.683**	**0.671**	0.631	0.627
+extdef +syn +hyper +hypo	0.680	0.661	0.632	0.621
Average	0.579	0.576	0.560	0.566

Summary of Results over the Example Sentences

- Definition expansion via the ontology produces significant performance gains (esp. hyponmys)

- Definition-level expansion has little impact

- Sense information helps out a bit ($\approx$4% absolute increment)

- Little difference between character and word tokenisation (other than for most basic methods)

- Best results better than Banerjee and Pedersen, 2003 and first-sense baseline

Breakdown across Word Classes

		ALL	NOUN	VERB	ADJ	ADV
UNSUPERVISED (RANDOM) BASELINE:		0.527	0.641	0.252	0.415	0.564
WORD	simple	0.469	0.620	0.145	0.294	0.388
	+syn	0.560	0.679	0.281	0.420	0.609
	+hyper	0.559	0.679	0.281	0.384	0.609
	+hypo	0.656	0.747	0.432	0.571	0.645
	+hyper +hypo	0.648	0.739	0.423	0.553	0.653
	+syn +hyper +hypo	0.650	0.743	0.419	0.615	0.665
	+extdef	0.489	0.630	0.179	0.306	0.451
	+extdef +syn	0.577	0.717	0.282	0.315	0.590
	+extdef +hyper	0.577	0.717	0.282	0.380	0.590
	+extdef +hypo	0.653	0.741	**0.434**	0.584	0.664
	+extdef +hyper +hypo	**0.683**	**0.789**	0.429	0.574	0.644
	+extdef +syn +hyper +hypo	0.680	0.785	0.428	**0.619**	**0.659**

Summary of Results for different POS

- Verbs (as always) a hard nut, but also the word class that benefits most from the proposed method

- Hyponyms particularly effective for verb and adjective WSD

Results over Senseval-2

	Sense-sensitive		Sense-insensitive	
	Word	Char	Word	Char
Unsupervised (random) baseline:		0.310		
Supervised (first-sense) baseline:		0.577		
simple	0.404	0.373	0.404	0.341
+extdef	0.420	0.362	0.420	0.329
+hyper	0.441	0.450	0.425	0.426
+hypo	0.568	0.577	**0.616**	0.610
+hyper +hypo	0.585	0.591	0.596	0.608
+extdef +hyper	0.451	0.484	0.371	0.432
+extdef +hypo	0.616	**0.630**	0.610	**0.622**
+extdef +hyper +hypo	**0.624**	0.624	0.593	0.602
Average	0.514	0.511	0.504	0.496

Summary of Results over Senseval-2

- Same basic trends as for the Hinoki example sentence data (but less increment for sense-sensitivity)

- Points of comparison:
  - ⋆ best result (0.630) $\equiv$ E.R.R. of 11.1%, c.f. E.R.R. of 21.9% for the best of the (supervised) WSD systems in the original Senseval-2 task

Miscellaneous Reflections

- Ontology has a big impact on results (esp. homonyms)
- Impact of sense-sensitivity (i.e. large-scale sense annotation) slight but appreciable
- Little to separate characters from words
- We blurr the boundary between unsupervised and supervised WSD somewhat in using the sense annotations in the definition sentences

Other Miscellaneous Results

- Tried segment weighting (TF·IDF), but it had very little impact
- Tried segment bigrams vs. unigrams, but unigrams tended to work better
- Tried stop word filtering (specific to dictionary domain), but it had little impact
- Tried applying POS constraints, but they had little impact on results

Applications of WSD

- WSD is all well and good, but ...
- Murmurings in recent years in the WSD community about what's it all about ...
- Notable successes of WSD in broader context: SMT, Penn Treebank parsing
- Applications of Hinoki WSD:
  - ⋆ parse selection (Fujita et al., 2007)
  - ⋆ context-sensitive glossing (Yap and Baldwin, 2007)

Imagine ...

答えなんてない　誰も教えてくれない
もしどこかにあるとしたら　君はもう手にしてる
貫くって決めたんなら　思いきり胸張って
顔を上げる事

他の誰かと君を比べてみたところで
基準が違うし何の意味も無い

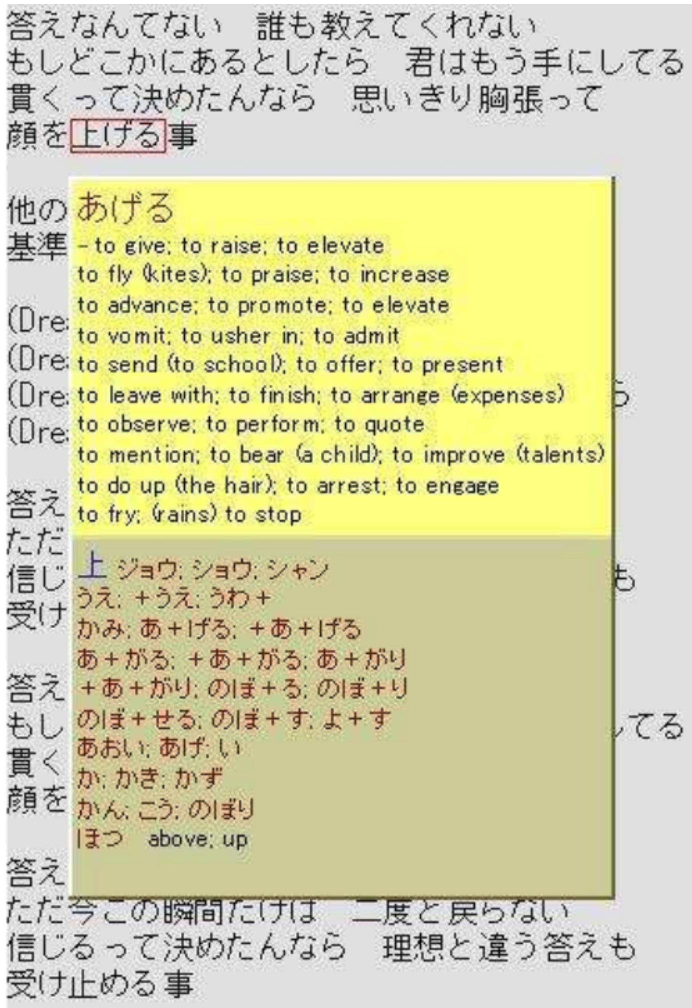
(Dreamin’ Startin’)君はこの世にひとり
(Dreamin’ Startin’)君の代わりはいない
(Dreamin’ Startin’)それでもためらうんなら
(Dreamin’ Startin’)それこそ君次第

答えなんてない　そんなのどこにもない
ただ今この瞬間だけは　二度と戻らない
信じるって決めたんなら　理想と違う答えも
受け止める事

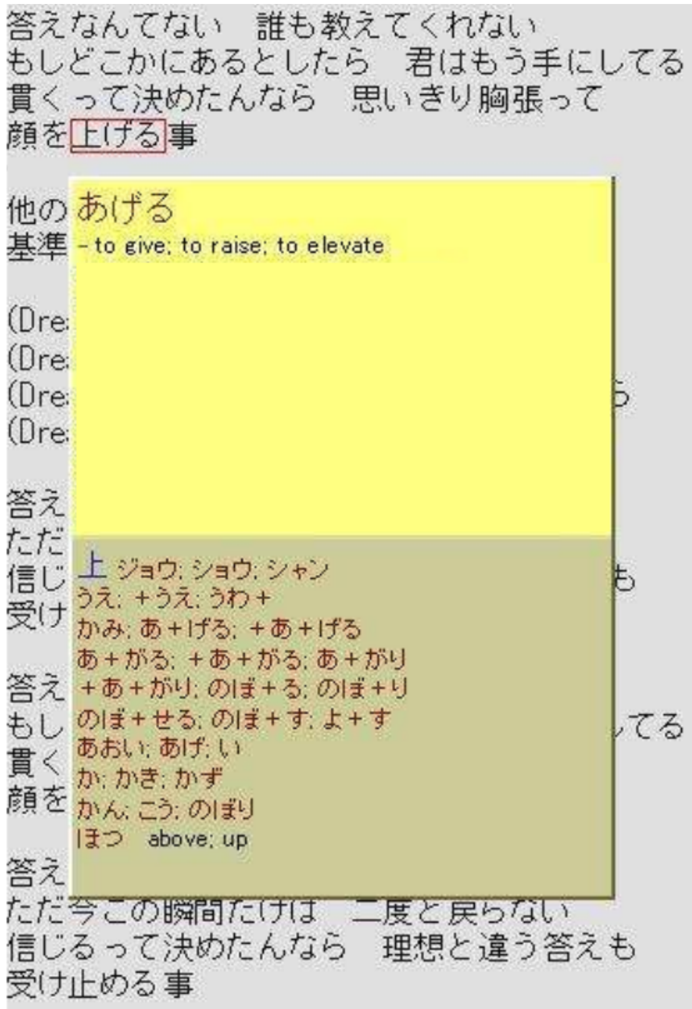
答えなんてない　誰も教えてくれない
もしどこかにあるとしたら　君はもう手にしてる
貫くって決めたんなら　思いきり胸張って
顔を上げる事

答えなんてない　そんなのどこにもない
ただ今この瞬間だけは　二度と戻らない
信じるって決めたんなら　理想と違う答えも
受け止める事

The Rikai Solution



The Rikai + WSD Solution



Context-sensitive Glossing

- On-line glossing an effective tool for people with incomplete knowledge of the language who can piece together an interpretation based on linguistic fragments

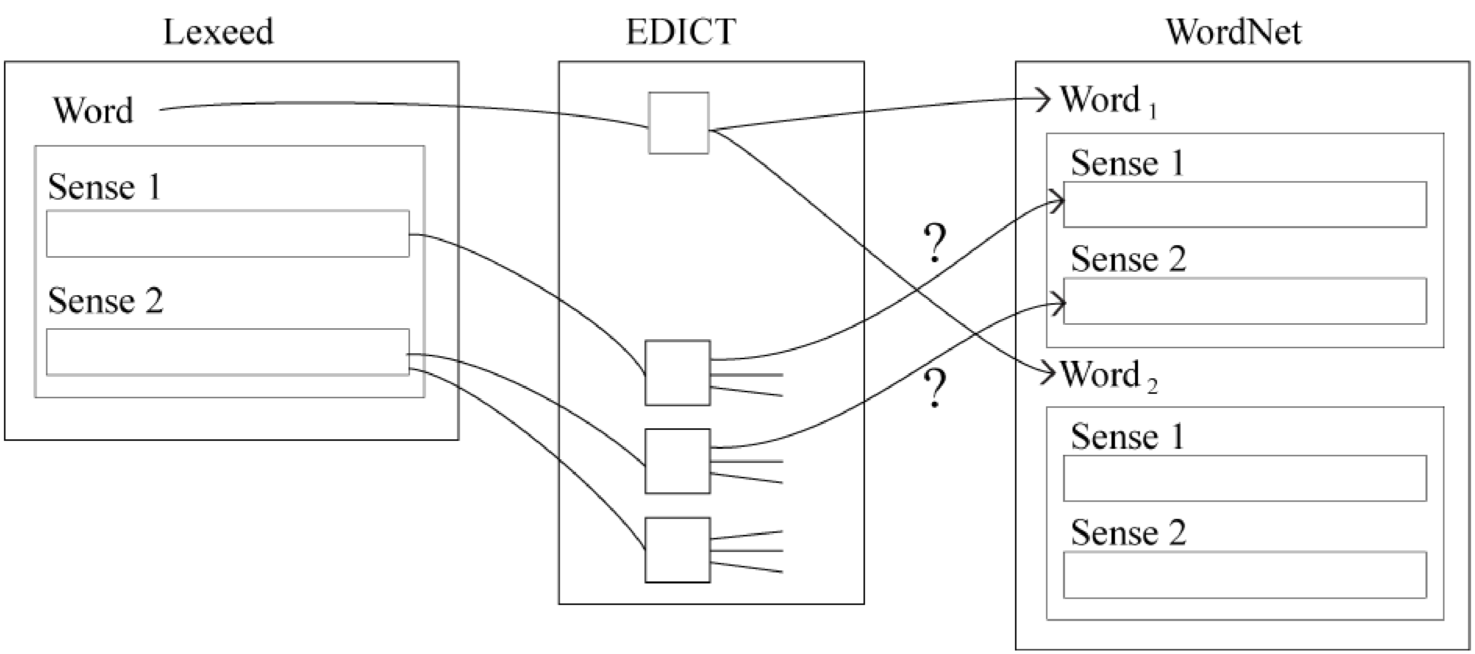
- BUT current online applications suffer from lack of context sensitivity

- Our solution:

  1. WSD the target text
  2. map the Japanese sense predictions onto English glosses via a dictionary alignment

Alignment Process

- Use EDICT as a pivot to:
  1. match Lexeed and WordNet head words
  2. match Lexeed and WordNet definitions



Reflections on Glossing

- Nice application of WSD in real-world context (with real-world users)

- Most effort to date has been expended on dictionary alignment

- Novel application where:
  - ⋆ best-1 sense disambiguation not necessarily required (appropriate balance of precision vs. recall, given "expert" post-editting)
  - ⋆ perfect dictionary sense alignment not necessary

- Will become considerably easier with Japanese WordNet (please, please, please) ...

Conclusion

- Development of simple baseline WSD methods/results for the Lexeed data to calibrate future experiments against
- Finding that ontological semantics in dictionary definitions leads to significant increments in WSD performance
- Ongoing exploration of applications of WSD in glossing context

References

Banerjee, S. and Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-2003)*, pages 805–810, Acapulco, Mexico.

Fujita, S., Bond, F., Oepen, S., and Tanaka, T. (2007). Exploiting semantic information for HPSG parse selection. In *Proceedings of the ACL 2007 Workshop on Deep Linguistic Processing*, pages 25–32, Prague, Czech Republic.

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 1986 SIGDOC Conference*, pages 24–26, Ontario, Canada.

Yap, W. and Baldwin, T. (2007). Dictionary alignment for context-sensitive word glossing. In *Proceedings of the Australasian Language Technology Workshop 2007 (ALTW 2007)*, pages 125–133, Melbourne, Australia.