Distributional Semantic Models for Affective Text Analysis and Grammar Induction

Alexandros Potamianos

School of ECE, National Technical Univ. of Athens, Greece

Acknowledgements

- Elias Iosif, Kelly Zervanou, Georgia Athanasopoulou: semantic similarity computation, semantic networks, semantic spaces
- Nikos Malandrakis, Shri Narayanan (USC): affective models for text, dialogue and multimedia
- Giannis Klassinas, Georgia Athanasopoulou, Elias Iosif, Spyros Georgiladakis, Elissavet Palogiannidi: grammar induction for spoken dialogue systems

References

[1] E. Iosif and A. Potamianos. 2010. “Unsupervised semantic similarity computation between terms using web documents”. IEEE Transactions on Knowledge and Data Engineering.
[2] E. Iosif and A. Potamianos. 2013. “Similarity computation using semantic networks created from web-harvested data”. Natural Language Engineering.
[3] N. Malandrakis, A. Potamianos, E. Iosif and S. Narayanan. 2013. “Distributional Semantic Models for Affective Text Analysis”. IEEE Transactions on Audio, Speech and Language Processing.
[4] K. Zervanou, E. Iosif and A. Potamianos. 2014. “Word Semantic Similarity for Morphologically Rich Languages”. In Proc. LREC.
[5] N. Malandrakis et al. 2014. “Affective language model adaptation via corpus selection”. In Proc. ICASSP.
[6] G. Athanasopoulou, E. Iosif and A. Potamianos. 2014. “Low-Dimensional Manifold Distributional Semantic Models”. Submitted to COLING.
[7] S. Georgiladakis et al. 2014. “Fusion of knowledge-based and data-driven approaches to grammar induction". Submitted to Interspeech.

Talk Outline

- Motivation: Cognitive Semantic Models
- Semantic similarity estimation
  - Web data harvesting
  - Network-based Distributional Semantic Models (DSMs)
  - Hierarchical manifold DSMs
- Semantic-affective models of text
  - Affective lexica and semantic-affective maps
  - Compositional semantics and affect
  - Affective model adaptation
- PortDial and SpeDial project overview
  - Grammar induction
  - Web data harvesting
  - SemEval 2014 task on grammar induction

List of Open Questions

1. How are concepts, features/properties, categories, actions represented?
2. How are concepts, properties, categories, actions combined (compositionally)?
3. How are judgements (classification/recognition decisions) achieved?
4. How is learning and inference (especially induction) achieved?

Answers should fit evidence by psychology and neurocognition!

Three Solutions

- Symbolic
  - cognition is a Turing machine
  - computation is symbol manipulation
  - rule-based, deterministic (typically)
- Associationism, especially, connectionism (ANNs)
  - brain is a neural network
  - computation is activation/weight propagation
  - example-based, statistical, unstructured (typically)
- Conceptual
  - intermediate between symbolic and connectionist
  - concepts are represented as well-behaved (sub-)spaces
  - computation tools: similarity, operators, transformations
  - hierarchical, semi-structured

Properties of the Three Approaches

- Symbolic
  - Good for high-level cognitive computations (math)
  - Poor generalization power
  - Too expensive and slow for most cognitive purposes
- Conceptual
  - Excellent generalization power (intuition, physics)
  - Good for induction and learning; geometric properties (hierarchy, low dim., convex) guarantee quick convergence
  - Properties and actions defined as operators/translations
  - Still too slow for some survival-dependent decisions
- Connectionist (machine learning)
  - General-purpose, extremely fast and decently accurate
  - Computational sort-cuts create cognitive biases
  - Poor generalizability power due to high dimensionality and lack of crisp semantic representation

Representation Learning

- Properties of a classifier with good generalization properties [Bengio et al 2013]:
  - Low-dimensionality/Sparseness
  - Distributed representations/hierarchy
    - Depth and abstraction
  - Shared factors across tasks
- Examples: auto-encoders, manifolds, deep neural nets ...
- How to induce these properties in your classifiers:
  - Include as regularization term in training classifier criterion
  - Include properties directly in classifier design
  - Go deep and pray (dirty neural net tricks)

Our Vision

- Cognitively-motivated semantic models
  - Emphasis on induction not classification
  - Associations not probabilities/distance
  - Mappings between layers
  - Hierarchical manifold models not metric spaces
  - Multimodal not unimodal
  - Other cognitive considerations ...

Problem Definition

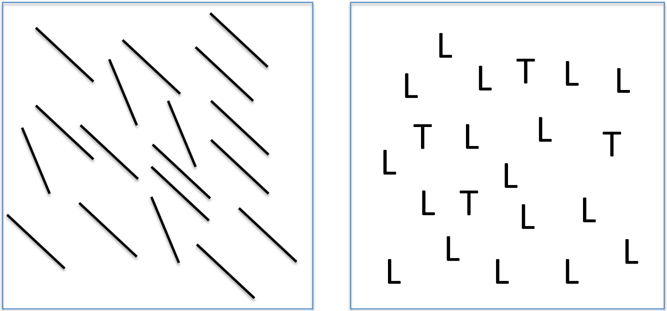
- Semantic Similarity Computation
  - Given a pair of words or terms (w_i, w_j)
  - Compute semantic similarity between them $S(i, j)$
- Related tasks
  - Phrase or sentence level semantic similarity
  - Strength of associative relation between words
  - Affective score (valence) of words and sentences
- Motivation
  - Organizing principle of human cognition
  - Building block of machine learning in NLP/semantic web
  - Entry point for the semantics of language

System 1 vs System 2

- Using Kahneman's (and others) formalism:
  - System 1 (intuition): generates
    - impressions, feelings, and inclinations
  - System 2 (reason): turns System 1 input into
    - beliefs, attitudes, and intentions
- Associative relations reside in System 1
- But where do semantic relations reside?

Example

- Example from vision: system 1 vs system 2

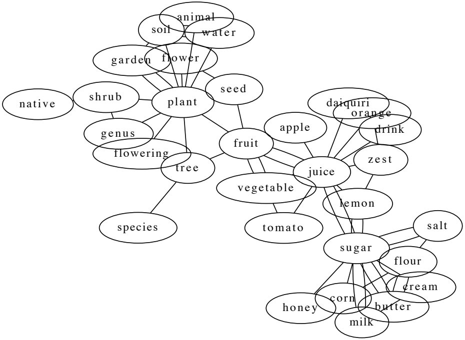


Main approaches of lexical semantics

- Word are associated with feature vectors
  - crisp, parsimonious representation of semantics
- Distributional semantic models (DSMs)
  - Semantic information extracted from word frequencies
  - Estimate co-occurence counts of word pairs or triplets
  - Estimate statistics of word context vectors
- Semantic networks
  - discovery of new relations via systematic co-variation
  - robust estimates – smoothing corpus statistics over network
  - rapid language acquisition

Example of Semantic Network

- Linked nodes: lexicalized senses and attributes
  - Informative for semantic similarity computation
- Computation of structural properties, e.g., cliques



Proposed semantic similarity two-tier system

- Unifies the three approaches
- Fuzzy vs explicit semantic relations
- Word senses vs words vs concepts
- A two tier system
  - An associative network backbone
  - Semantic relations defined as operations on network neighborhoods (cliques)
- Consistent with system 1 vs system 2 view
- Furthermore we believe that the
  - underlying network consists of word senses, and
  - is a low dimensional semi-metric space

Semantic Similarity Estimation by Machines

- Resource-based, e.g., WordNet
  - Require expert knowledge
  - Not available for all languages
- Corpus-based
  - Distributional semantic models (DSMs)
  - Unstructured (unsupervised): no use of linguistic structure
  - Structured: use of linguistic structure
  - Pattern-based, e.g., Hearst patterns
- Mixed

Semantic Sim. Computation: Sense Similarity

- Maximum sense similarity assumption *[Resnik, '95]*:
  - Similarity of words equal to similarity of their closest senses
  - If words are considered as sets of word senses, this is the "common sense" set distance
- Given words w_1, w_2 with senses s_{1i}, s_{2j}

$$S(w_1, w_2) = \max_{ij} S(s_{1i}, s_{2j})$$

Semantic Sim. Computation: Attributional Similarity

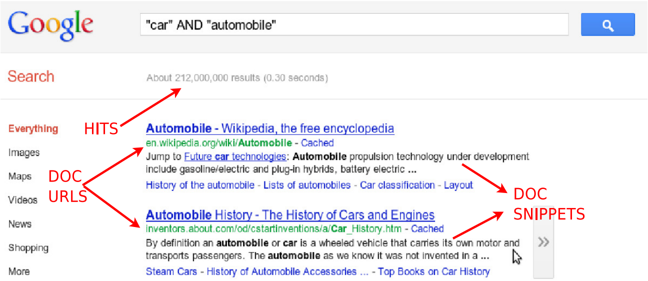
Attributional similarity assumption

- Attributes (features) reflect semantics
  - Item-Relation-Attribute, e.g., canary-color-yellow
- Main representation schemes
  - Hierarchical/Categorical
    - Mainly taxonomic relations, e.g., IsA, PartOf
  - Distributed (networks)
    - Open set of relations, e.g., Cause-Effect, etc
- Similarity between words
  - Function of attribute similarity
  - Defined wrt representation

Types of Similarity Metrics

- Co-occurrence-based
  - Assumption: co-occurrence implies relatedness
  - Co-occurrence counts: web hits, corpus-based
  - Examples: Dice coef., point-wise mutual information, ...
- Context-based
  - Assumption: context similarity implies relatedness (distributional hypothesis of meaning)
  - Contextual features extracted from corpus
  - Examples: Kullback-Leibler divergence, cosine similarity, ...
- Network-based (proposed)
  - Build lexical net using co-occurrence and/or context sim.
  - Notion of semantic neighborhoods
  - Assumptions: neighborhoods capture word semantics

Queries to Web Search Engines



- Number of hits
- Document URLs (download)
- Document snippets

Corpus Creation using Web Queries

- Two types of web queries
  - AND, e.g., "money + bank"
    "... leading **bank** in India offering online **money** transfer ..."
  - IND, e.g., "bank"
    "... downstream parallel to the **banks** of the river ..."
- AND queries
  - Pros: Similarity computation highly correlated (0.88) with human ratings *[Iosif & Potamianos, '10]*
  - Cons: Quadratic query complexity wrt lexicon *L*
- IND queries
  - Pros: Linear query complexity wrt lexicon *L*
  - Cons: Sense ambiguity: moderate correlation (0.55)

Semantic Similarity Estimation

- Co-occurence based metrics
  - From web: hits of IND, AND queries
  - From (web) corpus: co-occurence counts at the snippet or sentence level
  - Metrics: Dice, Jacard, Mutual Information, Google
- Context-based metrics
  - Download a corpus of documents of snippets using IND queries
  - Construct lexical context vector for each word (window ± 1)
  - Cosine similarity using binary or log-weighted counts

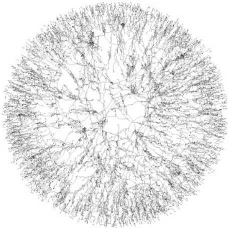
Enter semantic networks

- Why do IND queries fail to achieve good performance?
  1. Word senses are often semantically diverse
     - co-occurrence acts as a semantic filter
  2. Word senses have poor coverage in IND queries
     - rare word senses of words not well-represented
- Solution: use semantic networks
  1. Create a corpus for all words in lexicon (not just semantic similarity pair)
  2. Use semantic neighborhoods for semantic cohesion
     - improved robustness
  3. Inverse frequency word-sense discovery
     - discover rare senses via co-occurrence with infrequent words

Corpus and Network Creation

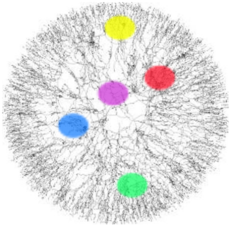
- Goals
  - Linear web query complexity for corpus creation
  - New similarity metrics with high performance
- Proposed method
  - IND queries to aggregate data for large L ($\approx 9K$)
  - Create network and semantic neighborhoods
  - Neighborhood-based similarity metrics
- Advantages
  - Network: parsimonious representation of corpus statistics
  - Smooth distributions
  - Rare words: well-represented
  - Enable discovery of less frequent senses

Lexical Network - Semantic Neighborhoods



Lexical Network

- Undirected graph $G = (N, E)$
  - Vertices N: words in lexicon L
  - Edges E: word similarities



Semantic Neighborhoods

- For word i create subgraph G_i
- Select neighbors of i
  - Compute $S(i, j), \forall\, j \in L, i \neq j$
  - Sort j according to $S(i, j)$
  - Select $|N_i|$ top-ranked j

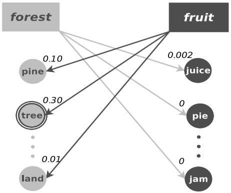
Semantic Neighborhoods: Examples

Word	Neighbors
automobile	**auto**, truck, **vehicle**, **car**, **engine**, bus, ...
car	truck, **vehicle**, travel, service, **price**, **industry**, ...
slave	slavery, beggar, **nationalism**, society, **democracy**, **aristocracy**, ...
journey	**trip**, holiday, **culture**, **travel**, **discovery**, **quest**, ...

- Synonymy
- Taxonomic: IsA, Meronymy
- Associative
- Broader semantics/pragmatics
- ...

Neighborhood-based Similarity Metrics: M_n

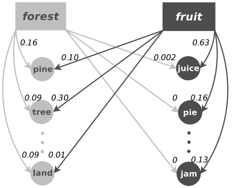
M_n metric: maximum similarity of neighborhoods



- Motivated by maximum sense similarity assumption
  - Neighbors are semantic features denoting senses
  - Similarity of two closest senses
- Select max. similarity: $M_n(\text{“forest”}, \text{“fruit”}) = 0.30$

Neighborhood-based Similarity Metrics: R_n

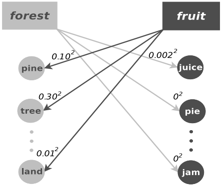
R_n metric: correlation of neighborhood similarities



- Motivated by attributional similarity assumption
  - Neighborhoods encode word attributes (or features)
  - Similar words have co-varying sim. wrt their neighbors
- Compute correlation r of neighborhood similarities
  - $r_1((0.16 \ldots 0.09), (0.10 \ldots 0.01)), r_2((0.002 \ldots 0), (0.63 \ldots 0.13))$
- Select max. correlation: $R_n(\text{"forest"}, \text{"fruit"}) = -0.04$

Neighborhood-based Similarity Metrics: metric $E_n^{\theta=2}$

$E_n^{\theta=2}$ metric : sum of squared neighborhood similarities



- Motivation: middle road between M_n and R_n
  - Accumulation of word–to–neighbor similarities
  - Non-linear weighting of similarities via $\theta = 2$
- $E_n^{\theta=2}$(“forest”, “fruit”)= $\sqrt{(0.10^2 + \cdots + 0.01^2) + (0.002^2 + \cdots + 0^2)} = 0.22$

Performance of net-based similarity metrics

- Task: similarity judgment on noun pairs
- Dataset: MC *[Miller and Charles, 1998]*
- Evaluation metric: Pearson's correlation wrt to human ratings

Dataset	Neighbor selection	Similarity computation	Metrics $M_{n=100}$	Metrics $R_{n=100}$	Metrics $E^{\theta=2}_{n=100}$
MC	co-occur.	co-occur.	0.90	0.72	**0.90**
MC	co-occur.	context	**0.91**	0.28	0.46
MC	context	co-occur.	0.52	**0.78**	0.56
MC	context	context	0.51	0.77	0.29

Main findings

- Network construction
  - Co-occurence metrics achieve high-recall for word senses
  - Context-based metrics achieve high-recall for attributes
- Semantic similarity performance
  - Co-occurence a more robust feature that context
  - Max sense similarity assumption is valid and gives best performance
  - Attributional similarity assumption valid for certain cases/languages

Performance of web-based similarity metrics

- For MC dataset

Feature	Description	Correlation
context	AND queries	0.88
context	IND queries	0.55
context	IND queries: network	0.90

- Comparable to structured DSMs, WordNet-based approaches

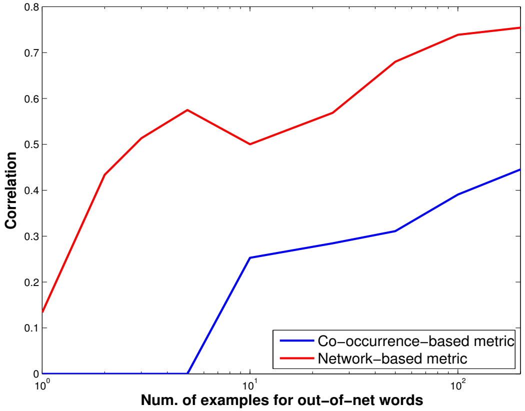
Extensions

- Sentence level semantic similarity (SemEval 2012, 2013)
- Abstract vs concrete semantic networks (IWSC 2013)
- Grammar induction in PortDial/SpeDial projects
- Morphologically rich languages (LREC 2014)
  - Network-based DSMs perform consistently well across languages
- Network DSMs and language acquisition (BabyAffect project)
  - Recognition vs generalization power (induction)
- Manifold DSMs
- Multimodal (text and image) conceptual spaces

Acquisition of lexical semantics

- Assume a recently acquired word *w*
  - Num. of *w*'s examples needed for "learning" *w*'s similarities
  - Related to acquisition of lexical semantics
- Compare
  - Simple co-occurrence-based similarity metric
  - Network-based similarity metric
- Experiment
  - 28 noun pairs (Miller-Charles dataset)
  - Remove one word from each pair from the network
  - Compute pair similarities
  - Evaluation: correlation coef. wrt human similarity ratings

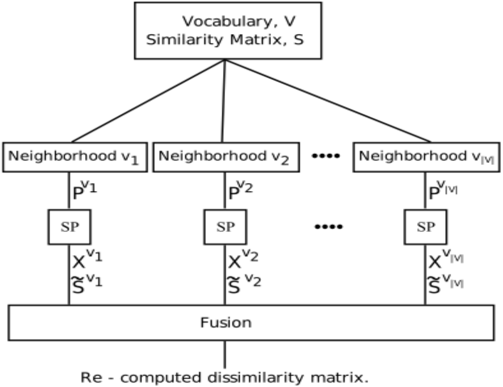
Acquisition of lexical semantics



Manifold DSMs

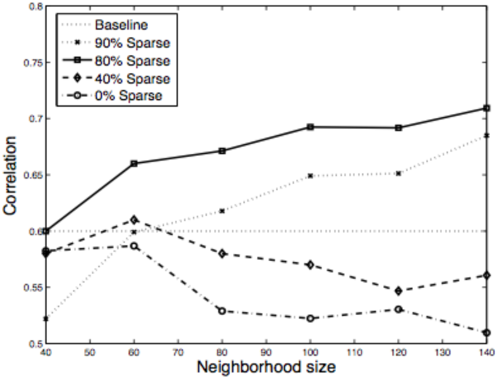
- Cognitive semantic space is fragmented in domains
- Sparse encoding of relations in each domain (manifold)
- Low-dimensional subspaces with good geometric properties
  - vs non-metric global semantic space
- Semantic similarity operation is performed locally in each subspace
- Decision fusion to reach semantic similarity score

Manifold DSMs

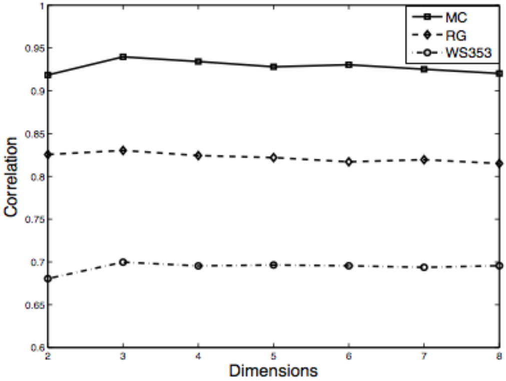


Sparse similarity matrices

- Correlation performance on the WS363 task



Effect of dimensionality



Contributions

Proposed a language agnostic, unsupervised and scalable algorithm for semantic similarity computation

- No linguistic knowledge required, works from text corpus or using a web query engine
- Shown to perform at least as well as resource-based semantic similarity computation algorithms, e.g., WordNet-based methods

Compositional Semantic-Affective Models of Text

Motivation

- Affective text labeling at the core of many multimedia applications, e.g.,
  - Sentiment analysis
  - Spoken dialogue systems
  - Emotion tracking of multimedia content
- Affective lexicon is the main resource used to bootstrap affective text labeling
  - Lexica are currently of limited scope and quality

Goals and Contributions

Our goal: assigning continuous high-quality polarity ratings to any lexical unit

- We present a method of expanding an affective lexicon, using web-based semantic similarity
- Assumption: semantic similarity implies affective similarity.
- The expanded lexica are accurate and broad in scope, e.g., they can contain proper nouns, multi-word terms

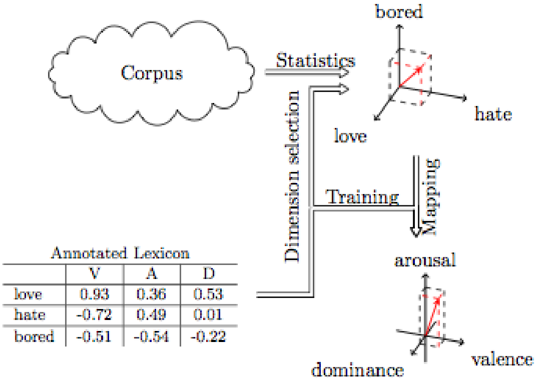
Our lexicon expansion method

Extension of [Turney and Littman, '02].
Assumption: the valence of a word can be expressed as a linear combination of the valence ratings of seed words weighted by semantic similarity and trainable weights a_i :

$$\hat{v}(t) = a_0 + \sum_{i=1}^{N} a_i \; v(w_i) \; d(w_i, t), \qquad (1)$$

- t : a word or n-gram (token) not in the affective lexicon
- $w_1 ... w_N$: seed words
- $v(.)$: valence rating of a word or n-gram
- a_i : weight assigned to seed w_i
- $d(w_i, t)$: semantic similarity between word w_i and token t

Semantic-Affective Mapping



	V	A	D
love	0.93	0.36	0.53
hate	-0.72	0.49	0.01
bored	-0.51	-0.54	-0.22

Given

- an initial lexicon of K words
- a set of $N < K$ seed words

we can use (1) to create a system of K linear equations with $N+1$ unknown variables:

$$\begin{bmatrix} 1 & d(w_1,w_1)v(w_1) & \cdots & d(w_1,w_N)v(w_N) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & d(w_K,w_1)v(w_1) & \cdots & d(w_K,w_N)v(w_N) \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} 1 \\ v(w_1) \\ \vdots \\ v(w_K) \end{bmatrix} \tag{2}$$

Solving with Least Mean Squares estimation provides the weights a_i.

Example, $N = 10$ seeds

Order	w_i	$v(w_i)$	a_i	$v(w_i) \times a_i$
1	mutilate	-0.8	0.75	-0.60
2	intimate	0.65	3.74	2.43
3	poison	-0.76	5.15	-3.91
4	bankrupt	-0.75	5.94	-4.46
5	passion	0.76	4.77	3.63
6	misery	-0.77	8.05	-6.20
7	joyful	0.81	6.4	5.18
8	optimism	0.49	7.14	3.50
9	loneliness	-0.85	3.08	-2.62
10	orgasm	0.83	2.16	1.79
-	w_0 *(offset)*	1	0.28	0.28

Sentence Tagging

Simple combinations of word ratings:

- linear (average)

$$v_1(s) = \frac{1}{N}\sum_{i=1}^{N} v(w_i)$$

- weighted average

$$v_2(s) = \frac{1}{\sum_{i=1}^{N} |v(w_i)|}\sum_{i=1}^{N} v(w_i)^2 \cdot \mathrm{sign}(v(w_i))$$

- max

$$v_3(s) = \max_i(|v(w_i)|) \cdot \mathrm{sign}(v(w_z)), \quad z = \arg\max_i(|v(w_i)|)$$

N-gram Affective Models

- Generalize method to n-grams

$$v_i(s) = a_0 + a_1 v_i(unigram) + a_2 v_i(bigram)$$

- Starting from all 1-grams and 2-grams, select terms:
  1. Backoff: use overlapping bigrams as default, revert to unigrams based on mutual information-based criterion
  2. Weighted interpolation: use all unigrams and bigrams as default, reject bigrams based on criterion
- In both cases unigrams and bigrams are given linear weights, trained using LMS

Evaluation

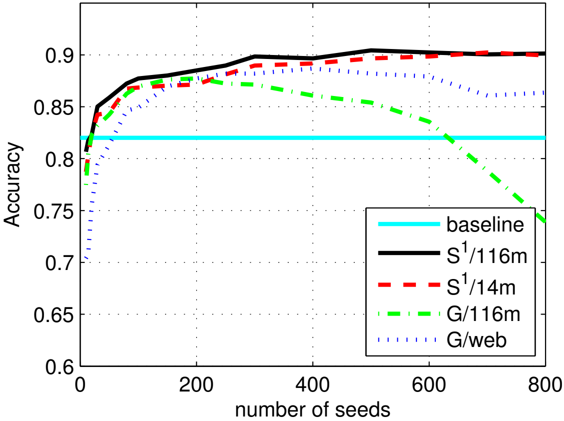
- ANEW Word Polarity Detection Task
  - Affective norms for English words (ANEW) corpus
  - 1.034 English words, continuous valence ratings
- General Inquirer Word Polarity Detection
  - General Inquirer words corpus
  - 3.607 English words, binary valence ratings
- BAWLR Word Polarity Detection Task
  - Berlin affective word list reloaded (BAWLR) corpus
  - 2.902 German words, continuous valence ratings
- SemEval 2007 Sentence Polarity Detection
  - SemEval 2007 News Headlines corpus
  - 1.000 English sentences, continuous valence ratings
  - ANEW used for lexicon training
  - 250 sentence development set used for word fusion training
- SemEval 2013, 2014: Twitter Sentiment Analysis

Experimental Procedure

- Corpus selection
  - Web corpus (web)
  - Lexically balanced web corpus (14m, 116m)
- Semantic Distance
  - Co-occurence based (G = google)
  - Context-based using web snippets (S)
- All experiments: training on ANEW seed words (cross-validation)

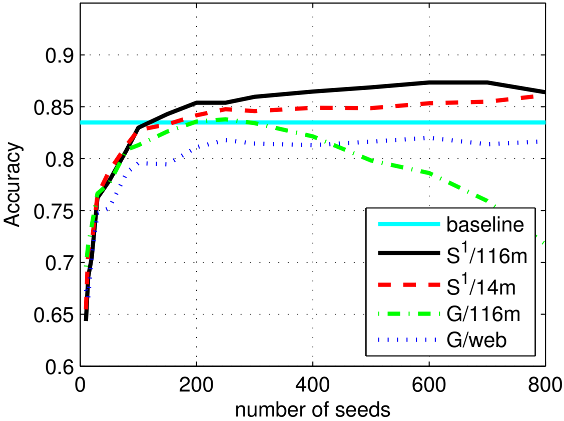
Word Polarity Detection (ANEW)

2-class word classification accuracy (positive vs negative)



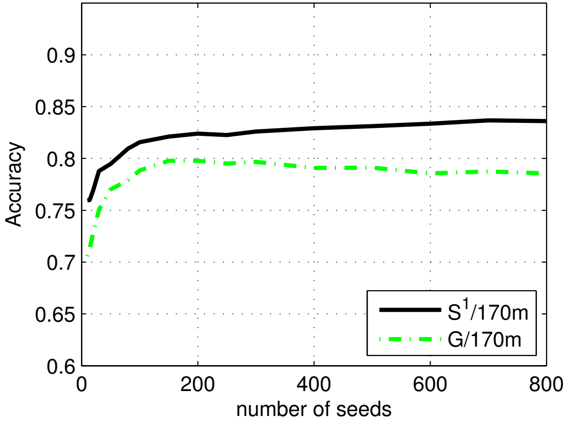
Word Polarity Detection (GINQ)

2-class word classification accuracy (positive vs negative)



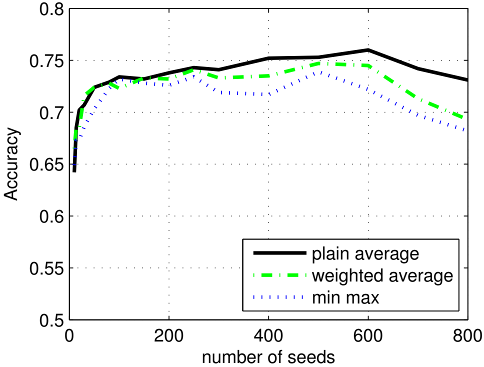
Word Polarity Detection (BAWLR)

2-class word classification accuracy (positive vs negative)



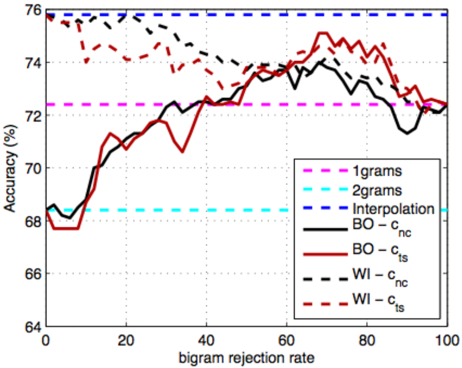
Sentence Polarity Detection (SemEval 2007)

2-class sentence classification accuracy (positive vs negative), using weighted interpolation



Sentence Polarity Detection (SemEval 2007)

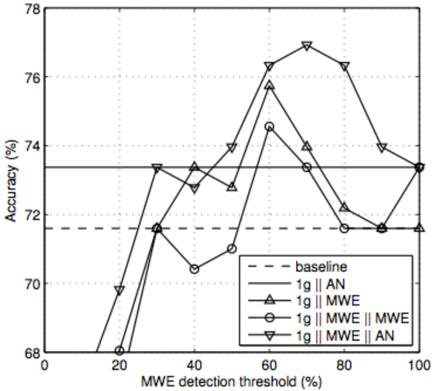
2-class sentence classification accuracy (positive vs negative), vs bigram rejection threshold



Sentence Polarity Detection (SemEval 2007)

- Using a compositional DSM model for AN pairs

2-class sentence classification accuracy

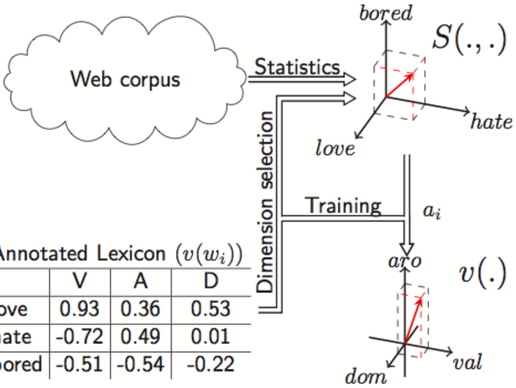


ChIMP Sentence Frustration/Politeness Detection

- ChIMP Children Utterances corpus
- 15.585 English sentences, Politeness/Frustration/Neutral ratings
- SoA results, binary accuracy P vs 0 / F vs O:
  - 81% / 62.7% [Yildirim et al, '05]
- 10-fold cross-validation
- ANEW used for training/seeds to create word ratings
- ChiMP words added to ANEW with weight *w*, to adapt to the task
- Similarity metric: Google semantic relatedness
- Only content words taken into account

Politeness: Sentence Classification Accuracy	Fusion scheme		
	avg	w.avg	max
Baseline: P vs O	0.70	0.69	0.54
Adapt $w = 1$: P vs O	0.74	0.70	0.67
Adapt $w = 2$: P vs O	0.77	0.74	0.71
Adapt $w = \infty$: P vs O	**0.84**	0.82	0.75
Frustration: Sentence Classification Accuracy	Fusion scheme		
	avg	w.avg	max
Baseline: F vs O	0.53	0.62	**0.66**
Adapt $w = 1$: F vs O	0.51	0.58	0.57
Adapt $w = 2$: F vs O	0.49	0.53	0.53
Adapt $w = \infty$: F vs O	0.52	0.52	0.52

Twitter Sentiment Analysis: Main Concept



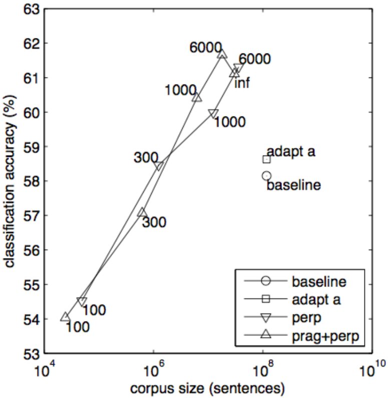
Annotated Lexicon ($v(w_i)$)	V	A	D
love	0.93	0.36	0.53
hate	-0.72	0.49	0.01
bored	-0.51	-0.54	-0.22

$$\underbrace{\underset{0.12}{\mathrm{I}}\ \underset{-0.71}{\mathrm{hate}}\ \underset{0.41}{\mathrm{you}}\ \underset{0.38}{\mathrm{so}}\ \underset{0.31}{\mathrm{much}}}$$

$$\text{VAD Statistics} \Rightarrow \text{Classifier} \Rightarrow \text{Anger} = \text{TRUE}$$

Twitter Sentiment Analysis: Semantic Adaptation

3-class sentence classification accuracy
(positive-neutral-negative) [ICASSP 2014]



SemEval 2014: Twitter Sentiment Results Analysis

Features removed	LJ2014		SMS2013		TW2013		TW2014		TW2014SC	
	avg. F1	diff	avg. F1	diff	avg. F1	diff	avg. F1	diff	avg. F1	diff
None (Submitted)	69.3		57.0		**66.8**		67.8		57.3	
Lexicon-derived	43.6	-25.8	38.2	-18.8	49.5	-17.4	51.5	-16.3	43.5	-13.8
Emotiword	67.5	-1.9	56.4		63.5	-3.3	66.1	-1.7	54.8	-2.5
Base	68.4		56.3		65.0	-1.9	66.4	-1.4	**59.6**	2.3
Adapted	69.3		57.4		66.7		67.5		50.8	-6.5
Sentiment140	68.1	-1.3	54.5	-2.5	64.4	-2.4	64.2	-3.6	45.4	-11.9
NRC Tag	**70.6**	1.3	**58.5**	1.6	66.3		66.0	-1.7	55.3	-2.0
SentiWordNet	68.7		56.0		66.2		**68.1**		52.7	-4.6
per Lexeme	69.3		56.7		66.1		68.0		52.7	-4.5
per Lexeme-POS	68.8		57.1		66.7		67.4		55.0	-2.2
Semantic Similarity	69.0		58.2	1.2	64.9	-2.0	65.5	-2.2	52.2	-5.0
Punctuation	69.7		57.4		66.6		67.1		53.9	-3.4
Emoticon	69.3		57.0		**66.8**		67.8		57.3	
Contrast	69.2		57.5		66.7		67.0		51.9	-5.4
Prefix	69.5		57.2		**66.8**		67.2		47.4	-9.9
Suffix	68.6		57.2		66.5		67.9		56.3	

Conclusions

Proposed a high-performing, robust, general-purpose and scalable algorithm for affective lexicon creation

- Investigated linear and non-linear sentence level fusion schemes, showing good but task-dependent performance
- Investigated domain adaptation: semantic space vs semantic-affective mapping adaptation
- Demonstrated that distributional approach can generalize to n-grams

Linguistic Resources for Spoken Dialogue Systems: The PortDial and SpeDial projects

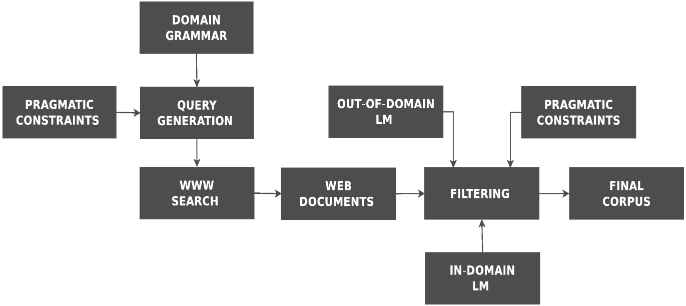
Outline

1. PortDial project
   - "Language Resources for Portable Multilingual Spoken Dialogue Systems"
   - 2-year EU-funded project: currently in last quarter
   - `www.portdial.eu`
2. SpedDial project
   - "Machine-Aided Methods for Spoken Dialogue System Enhancement and Customization for Call-Center Applications"
   - 2-year EU-funded project: currently in first quarter
   - `www.spedial.eu`
3. SemEval'14-Task 2
   - "Grammar Induction for Spoken Dialogue Systems"
   - Evaluation period: until March 30
   - `http://alt.qcri.org/semeval2014/task2/`

PortDial: Outline

- Grammars
  - Essential unit of spoken dialogue systems
  - Expertise needed, time-consuming
  - Need for rapid porting
- PortDial paradigm
  - Machine-aided process
  - Human-in-the-loop
- ProtDial approaches
  - Corpora creation via web harvesting
  - Grammar induction
    - Bottom-up: corpus-based
    - Top-down: ontology-based
    - Fusion of bottom-up and top-down

PortDial: Web-harvested corpora



- WWW search query, e.g., "depart from"&("flight"|"travel"|..)
- Out-of-domain LM: perplexity $\rightarrow$ grammaticality/spelling
- In-domain LM: perplexity $\rightarrow$ domain relevance

PortDial: Web-harvested corpora

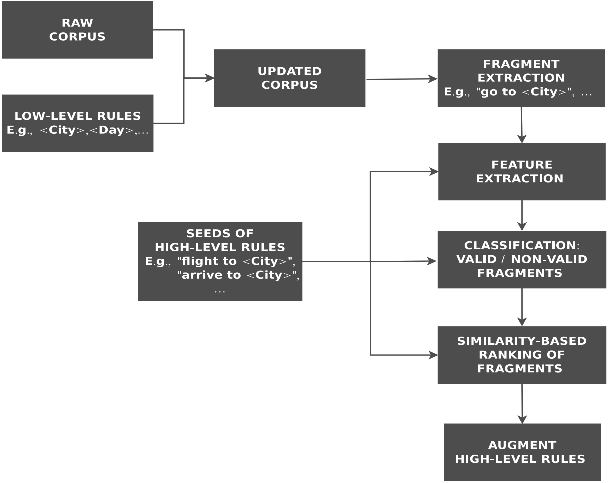
- Travel domain grammar
  - 83 low-level rules
    - E.g., <City> = (“New York”, “London”, ...)
  - 47 high-level rules
    - E.g., <ArrivalCity> = (“fly to <City>”, “arrive at <City>”, ...)
- Use of various corpora for inducing low-level rules

Corpus	Precision	Recall	F-measure
Q&A	0.52	0.40	0.45
WoZ	0.41	0.33	0.37
Human-Human	0.42	0.32	0.36
Human-System	0.41	0.34	0.37
Manually harvested	0.46	0.41	0.43
Web-harvested	0.56	0.45	0.50

PortDial: Bottom-up Grammar Induction

- Goal: induction of high-level rules
  - Based on the availability of low-level rules
- Minimal set of examples (seeds) are provided
  - Analogous to the manual process of grammar development
  - Examples are automatically augmented
- Two sub-problems
  1. Extraction of fragments from corpus
     - Retain fragments with appropriate boundaries
     - E.g., "to depart from <City> on" VS "depart from <City>"
  2. Similarity between seeds and extracted fragments
     - Retain semantically similar fragments
     - E.g., "out of <City>" VS "to <City>"

PortDial: Bottom-up Grammar Induction



PortDial: Bottom-up Grammar Induction

1. Extraction of fragments
   - Binary classification problem
     - Valid / non-valid fragments
     - Seeds considered as valid fragments
   - Types of features
     - Lexical, e.g., frequency in corpus, num. of tokens
     - Syntactic, e.g., fragment perplexity, PoS info.
     - Semantic, e.g., similarity wrt to seeds
2. Similarity computation
   - Non-compositional: fragments as entire chunks
     - Various well-known lexical metrics
     - E.g., longest common sub-string similarity
   - Compositional: function of constituents' semantics
     - Recent open research problem
     - Models proposed for sentences, but not for phrases

PortDial: Bottom-up Grammar Induction

- Evaluation
  - Travel domain
  - Input: n seeds for each rule
    - $n < 5$
  - Output: m fragments suggested for each rule
    - m: user-defined
- Accuracy
  - Valid / non-valid fragments classification: 43%
  - Suggestion of semantically similar fragments: 30%
- However, in practice
  - Some non-valid fragments may be useful
    - Lengthier, e.g., “depart from <City> on”
  - Human-in-the-loop idea
    - Post corrections
    - Iterative process

PortDial: Bottom-up & Top-down Grammar Induction

- Goal: Fuse different approaches for grammar induction
  - High-level rules
- Approaches
  1. Bottom-up: corpus-based
  2. Top-down: based on ontology lexica
- Bottom-up (BU)
  - Relies on given seeds for each grammar rule
  - Extraction and suggestion of similar textual fragments
- Top-down (TD)
  - Ontology lexica: lexicalizations of ontological knowledge
  - Represent domain semantics in ontological representation
  - Possible lexicalizations are encoded as grammar rules

PortDial: Bottom-up & Top-down Grammar Induction

- Three fusion approaches
  1. Early fusion
     - Rules of TD triggered to generate a corpus; input to BU
  2. Mid fusion
     - TD grammar rules given as seeds to BU
  3. Late fusion
     - Rules of TD and BU are combined (union)
- Evaluation: Travel domain

Approach	Precision	Recall	F-measure
Bottom-Up (BU)	0.65	0.44	0.52
Top-Down (TD)	0.81	0.18	0.30
Early fusion	0.64	0.44	0.52
Mid fusion	0.56	0.54	0.55
Late fusion	0.72	0.55	0.63

SpeDial Objectives

- Devise machine-aided algorithms for spoken dialogue system enhancement and customization for call-center applications
- Create a platform that supports cost-effective service doctoring for
  - Service enhancement: the developer starts from an existing application and tries to improve performance and user satisfaction,
  - Service customization: the developer addresses the special needs of a user population
- Create and support a sustainable pool of developers that will be trained to use the platform

SpeDial: Multimodal Analytics for IVR

- Affective analysis of dialogues
  - Valence, arousal, mood, certain/uncertain
  - Also: gender, age, nativeness identification
- Call-flow, discourse and cross-modal analytics
  - Identify problematic and successful parts of the dialogues
  - Identify dialogue hot-spots
- Multilingual analytics
  - How previous sub-tasks can be applied across multiple languages

SpeDial: Enhancement and Customization

- Prompt and grammar enhancement
  - Select most appropriate prompts from the pool of prompts
  - Use transcriptions to train/update statistical grammars
  - Update FSM grammars via grammar induction
- Dialogue flow enhancement
  - Adjustment of system policies for successful interactions
- User modeling: prompt selection wrt
  - Age, gender, age & gender
- Multilinguality
  - SMT & crowd-sourcing to improve on prompts & grammars
  - Corpus-based methods for statistical grammar training
  - Direct translation of service grammars

SemEval'14: Task on Grammar Induction

- SemEval workshops
  - Various shared evaluations tasks of computational semantic analysis systems
  - SemEval'14: the 8th workshop
  - Co-located with COLING'14, Dublin, Ireland, August 2014
- SemEval'14 - Task 2
  - "Grammar induction for spoken dialogue systems"
  - Fosters the application of models of lexical semantics to spoken dialogue systems
  - Organized by the consortium of PortDial project

SemEval'14: Task on Grammar Induction

- Grammar rules distinguished into
  1. Low-level
     - Refer to basic concepts; comprised by lexical items only
     - E.g., <City> = ("New York", "London", ...)
  2. High-level
     - Grouping of semantically related textual fragments
     - Composed of both lexical items and low-level rules
     - E.g., <ArrivalCity> = ("fly to <City>", "arrive at <City>", ...)
- Parsing example
  1. "I want to fly to London"
  2. "I want to fly to <City>"
  3. "I want to <ArrivalCity>"

SemEval'14: Task on Grammar Induction

- Sub-problems
  1. Induction of low-level rules
     1. Well-investigated
     2. Also, use of resources, e.g., gazetteers
  2. Induction of high-level rules
     1. Segmentation problem: identify candidate fragments
     2. Similarity problem: compute similarity between fragments
- SemEval'14-Task 2
  - Focus on high-level rules
    - Low-level rules are given
  - Segmentation problem simplified as:
    - Discriminate between valid / non-valid fragments
  - Main focus: computation of similarity between fragments

SemEval'14: Task on Grammar Induction

- Train data
  - List of fragments for each grammar rule
  - Instances of low-level rules: given
  - List of non-valid fragments
- Test data
  - List of unknown fragments
  - For each unknown fragment:
    1. Is it a valid fragment?
    2. If so, assign it to the most similar rule

Domain	Language
Travel	English
Travel	Greek
Tourism	English
Finance	English

Thank you