Aspect Based Sentiment Analysis

Ioannis Pavlopoulos

PhD student

Natural Language Processing Group

Department of Informatics

Athens **U**niversity of **E**conomics and **B**usiness

Task description

The food was delicious!
Nice food but horrible wine and beers.
Excellent service! Thank you ☺
…

ABSA

Aspect term extraction

food, wine, beers, service, …

Aspect term aggregation

food, wine, beers, service, …

Aspect term polarity

food, **wine**, **beers**, **service**, …

Visualisation

See our paper for more details (Pontiki et al., 2014)

Outline

1. Aspect term extraction
2. Multi-granular aspect aggregation
3. Message-level sentiment estimation
4. Aspect term sentiment estimation

Basis for 'aspect term extraction' ABSA SemEval 2014/2015 task

Previous datasets vs. our datasets

Datasets	Inter- Annotator Agreement	# Domains	Gold Aspect Terms
Hu & Liu 2004	✗	1	✓
Ganu et al. 2009	✓	1	✗
Blitzer et al. 2007	✗	4	✗
Pavlopoulos & Androutsopoulos 2014	✓	3	✓

Our new datasets

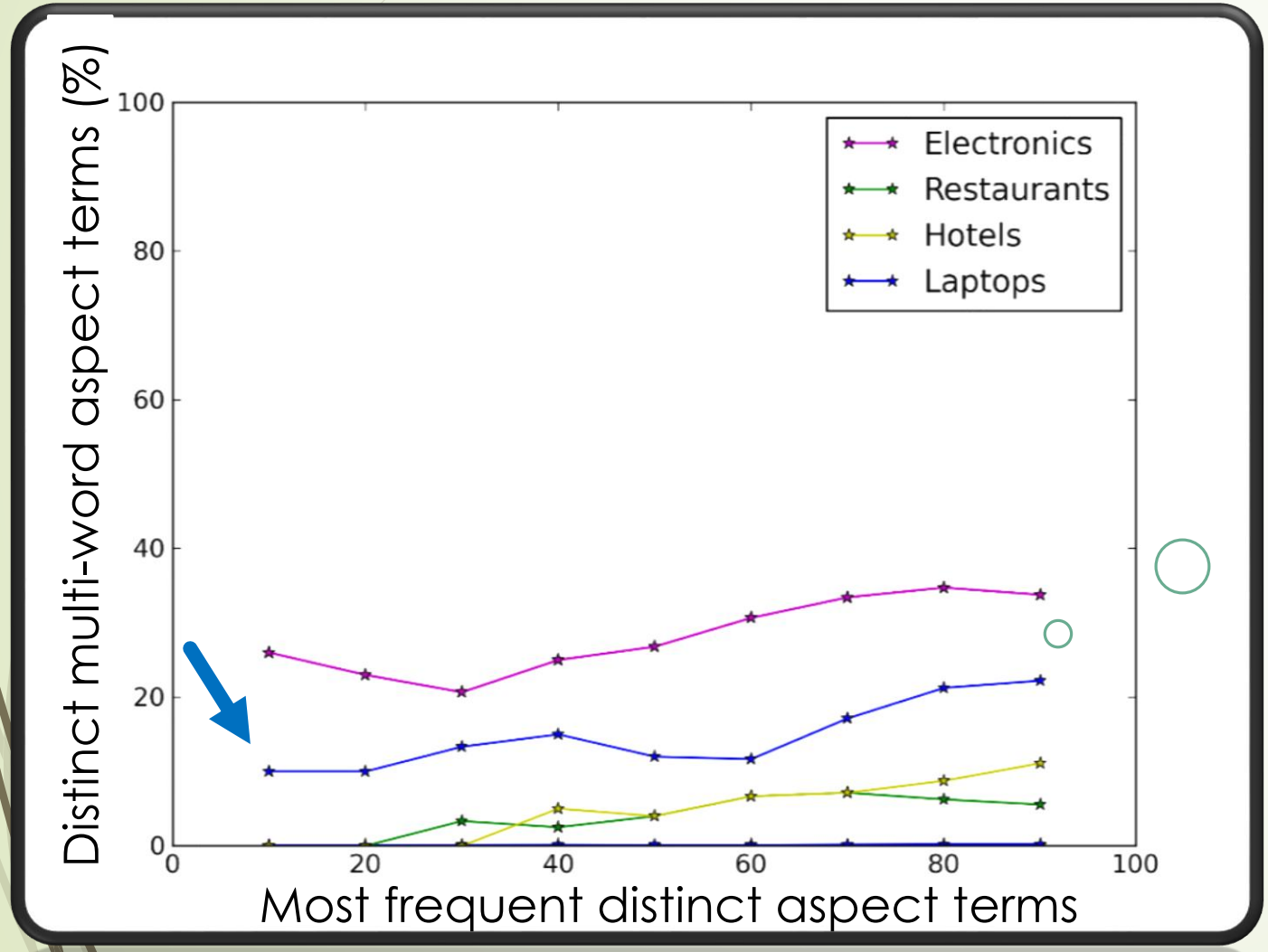
	# sentences with *n* aspect term occurrences			
Domain	*n* = 0	*n* > 0	*n* > 1	total
Restaurants	1,590	2,120	872	3,710
Hotels	1,622	1,978	652	3,600
Laptops	1,760	1,325	416	3,085

Inter-Annotator Agreement:
Dice: ~70% in all domains

battery life

	# distinct aspect terms with *n* occurrences			
	n > 0		*n* > 1	
Domain	*multi-word*	*single-word*	*multi-word*	*single-word*
Restaurants	**593**	452	67	**195**
Hotels	199	**262**	24	**120**
Laptops	**350**	289	67	**137**

Multi-word vs. single-word distinct aspect terms per domain



Electronics (Hu & Liu, 2004) & laptops reviews contain more multi-word distinct aspect terms

Precision, Recall, F-measure

Gold: “design” (94), “service” (3), “screen” (2)

Predicted: “design” (92/94), “service” (1/3 + 1), “screen” (0/2), “foo” (+3)

*Computed on **types** (**distinct** aspect terms):*

$$\mathbf{P} = \frac{|\text{true predicted } \boldsymbol{types}|}{|\text{predicted } \mathbf{types}|} = \frac{|\text{design,service}|}{|\text{design,service,foo}|} = \frac{\mathbf{2}}{\mathbf{3}} = 0.66$$

$$\mathbf{R} = \frac{|\text{true predicted } \boldsymbol{types}|}{|\text{true } \mathbf{types}|} = \frac{|\text{design,service}|}{|\text{design,service,screen}|} = \frac{\mathbf{2}}{\mathbf{3}} = 0.66$$

Frequent distinct aspect term are treated as rare ones

*Computed on **tokens** (aspect term **occurrences**):*

$$\mathbf{P} = \frac{|\text{true predicted } \boldsymbol{tokens}|}{|\text{predicted } \boldsymbol{tokens}|} = \frac{\mathbf{92+1+0+0}}{\mathbf{92+2+0+3}} = 0.96$$

$$\mathbf{R} = \frac{|\text{true predicted } \boldsymbol{tokens}|}{|\text{true } \boldsymbol{tokens}|} = \frac{\mathbf{92+1+0+0}}{\mathbf{94+3+2+0}} = 0.94$$

Over sensitive to high-frequent aspect terms

Precision, Recall, F-measure

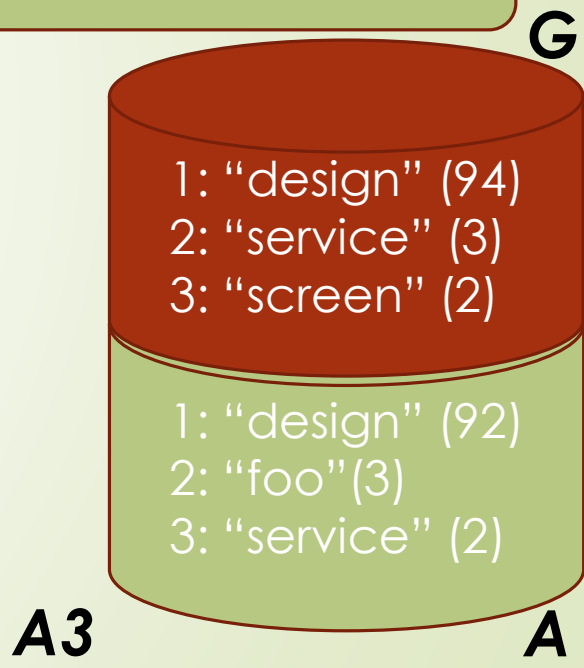
- **The users care only about the top m** (e.g., 10-20) **most frequently** discussed **distinct aspect terms**.
  - The value of m depends on **screen size**, **available time** etc.
- **Finding or missing a truly more frequent distinct aspect term** should be **rewarded or penalized more**.
- **Placing a truly high-frequency distinct aspect term** towards the **beginning of the returned list** should be **rewarded more**.

How we propose to measure

Gold: “design” (94), “service” (3), “screen” (2)

Predicted: “design” (92/94), “service” (1/3 + 1), “screen” (0/2), “foo” (+3)

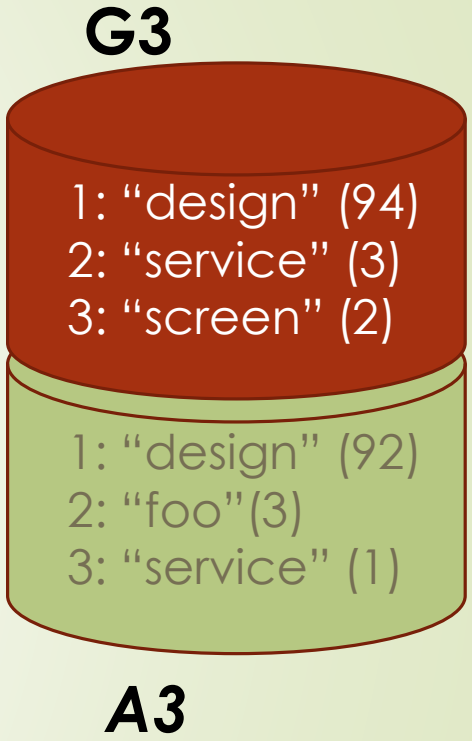
- Order all the correct distinct aspect terms by human annotation frequency (G list).
- Each method returns a list of distinct aspect terms, ordered by predicted frequency (A list).
  - Given an m value, use the first m elements of the A list (A_3).
- Compare G and A_m for different m values.



Weighted precision and recall

$$\mathbf{WP_m} = \frac{\sum_{i=1}^{m} \frac{1}{i} 1\cdot\{a_i \in G\}}{\sum_{i=1}^{m} \frac{1}{i}} \xrightarrow{m=3} \frac{\frac{1}{1} + 0 + \frac{1}{3}}{\frac{1}{1} + \frac{1}{2} + \frac{1}{3}} = 0.73$$

$$\mathbf{WR_m} = \frac{\sum_{i=1}^{m} \frac{1}{r(a_i)} 1\cdot\{a_i \in G\}}{\sum_{j=1}^{|G|} \frac{1}{i}} \xrightarrow{m=3} \frac{\frac{1}{1} + 0 + \frac{1}{2}}{\frac{1}{1} + \frac{1}{2} + \frac{1}{3}} = 0.82$$



- By **varying m**, we obtain $\boldsymbol{WP_m - WR_m}$ **curves**.
- Also, **average weighted precision** at 11 weighted recall levels.
- WP_m is similar to $nDCG@m$, but no counter-part for WR_m.

Methods

Freq baseline

- Considered effective & popular **unsupervised** baseline (Liu, 2012)
- Returns the **most frequent nouns and noun phrases**, ordered by decreasing sentence frequency

H&L (Hu & Liu 2004)

- Also **unsupervised**, finds **frequent nouns and noun phrases**, plus…
- **Discards candidate aspect terms** that are **subparts of other candidate aspect terms**
- Finds **adjectives that modify candidate aspect terms**, uses them to detect **additional candidate aspect terms**
- Details previously unclear, **full pseudo-code** published

Our pruning step

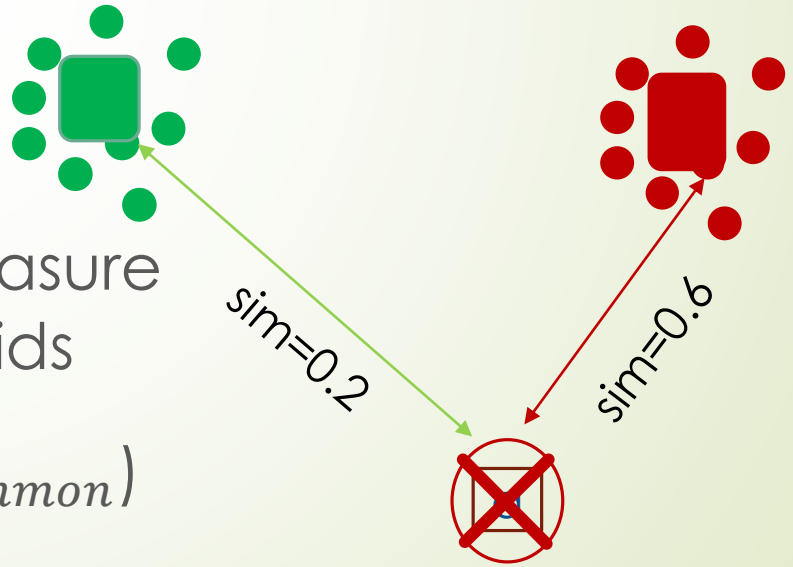
We use word vectors (Mikolov, 2013) computed using *Word2Vec*

$$v('king') - v('man') + v('woman') \cong v('queen')$$

$$v_{domain} = \frac{\sum_{w \in aspect\ terms} v(w)}{|aspect\ terms|}$$

$$v_{common} = \frac{\sum_{w \in common\ words} v(w)}{|common\ words|}$$

v('queen'): word vector of 'queen'
<0.2, 0.9, 0.0, …, 0.3, 0.7, 0.5>



For each candidate aspect term a, measure its similarity (cosine) with the two centroids

Prune a, if $\cos(a, v_{aspect}) < \cos(a, v_{common})$

Applicable to both ***Freq*** and ***H&L***

Results: average weighted precision

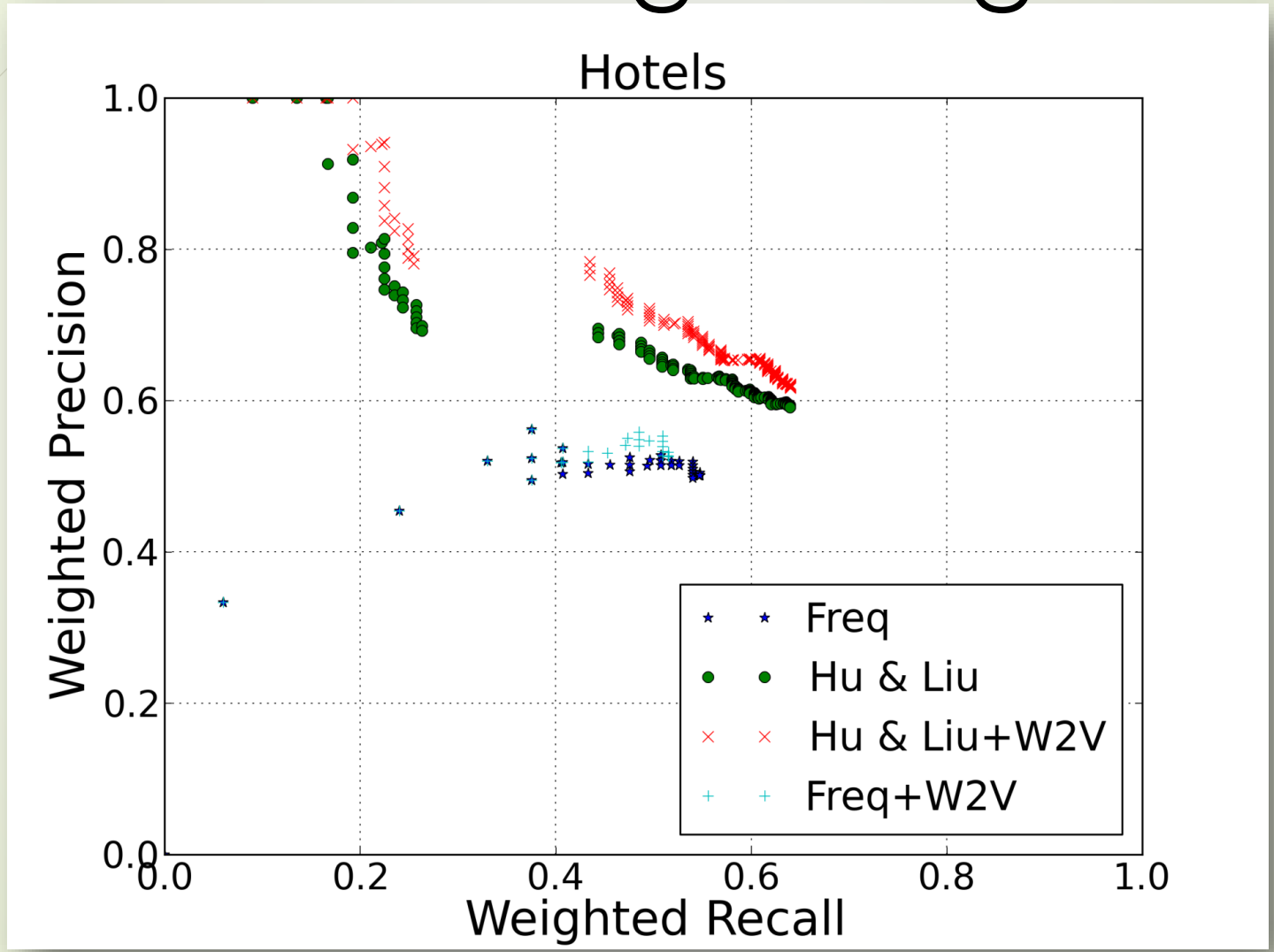
Method	Restaurants	Hotels	Laptops
Freq	43.40	30.11	**9.09**
Freq+w2v pruning	**45.17**	**30.54**	7.18

Method	Restaurants	Hotels	Laptops
Hu&Liu	52.23	49.73	34.34
H&L+w2v pruning	**66.80**	**53.37**	**38.93**

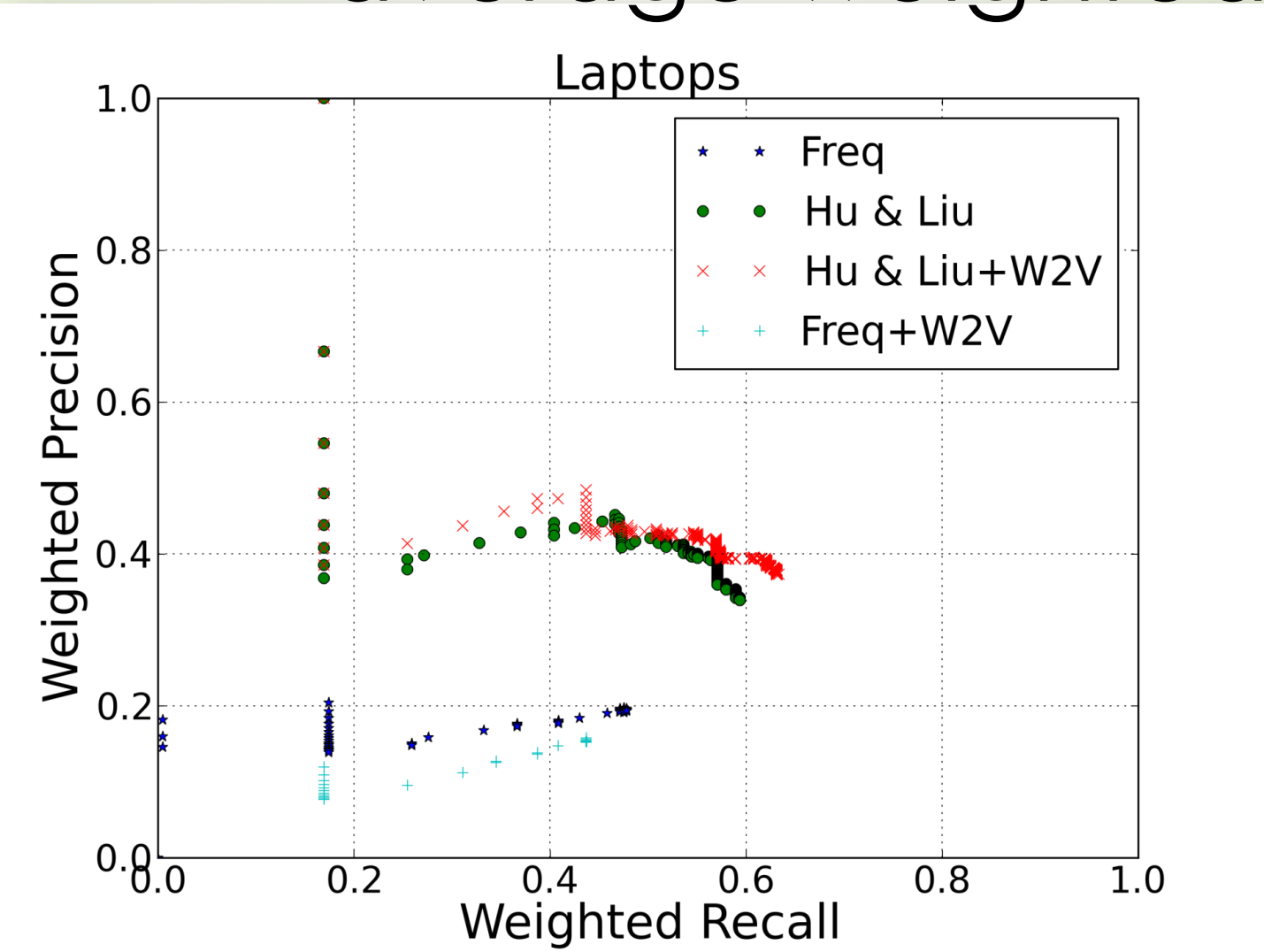
All differences are statistically significant ($p<0.01$)

Results: average weighted precision

Results: average weighted precision



Results: average weighted precision



See our paper for more details
(Pavlopoulos and Androutsopoulos, 2014a)

Summary & contribution of this section

- Introduced 3 new aspect term extraction datasets
  - Laptops/Restaurants/Hotels
  - Domain variety is important
- New evaluation measures
  - Weighted precision, weighted recall, average weighted precision
- Improved the popular unsupervised method of Hu & Liu
  - Additional pruning step based on continuous space word vectors (using Word2Vec)
- The 'Aspect term extraction' subtask of ABSA SemEval 2014 & 2015 was based on the work of this section

Outline

1. Aspect term extraction
2. Multi-granular aspect aggregation
3. Message-level sentiment estimation
4. Aspect term sentiment estimation

Task description

The food was delicious!
Nice food but horrible wine and beers.
Excellent service! Thank you ☺
…

ABSA

Aspect term extraction

food, wine, beers, service, …

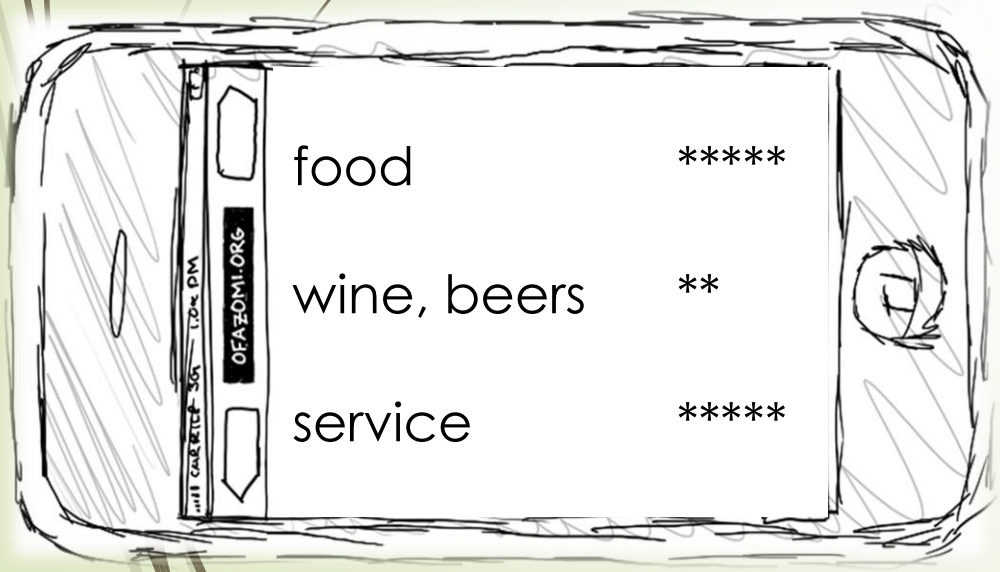
Aspect term aggregation

food, wine, beers, service, …

Aspect term polarity

food, **wine**, **beers**, **service**, …

Visualisation

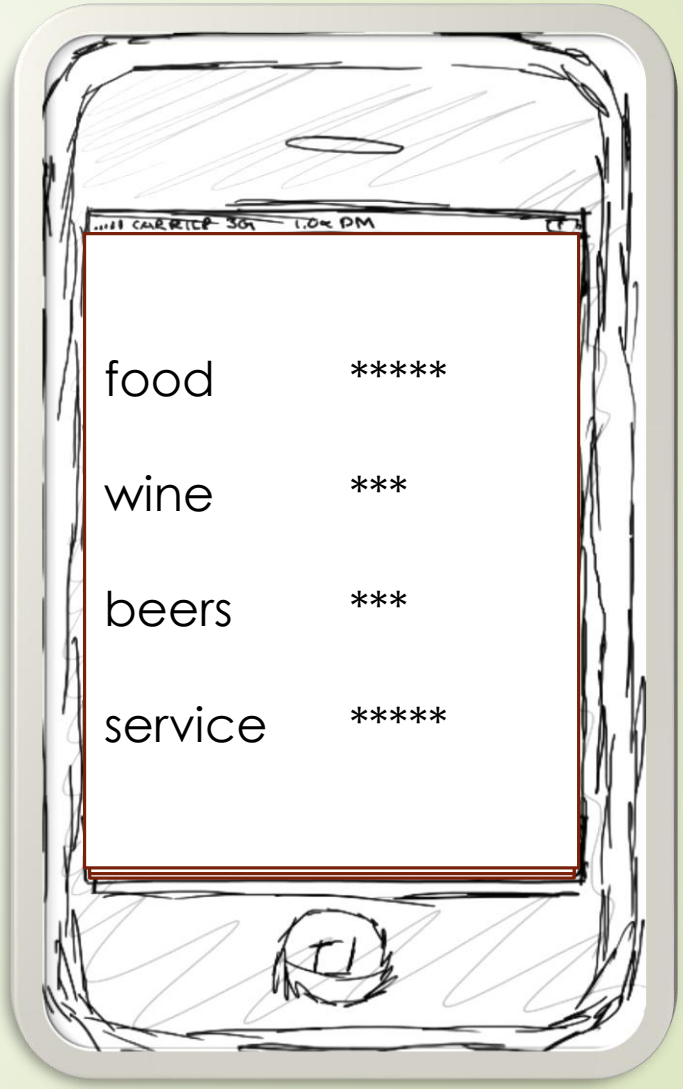


Aspect aggregation with multiple granularities

Top Aspect Terms

1. Food
2. Wine
3. Beers
4. Service

...



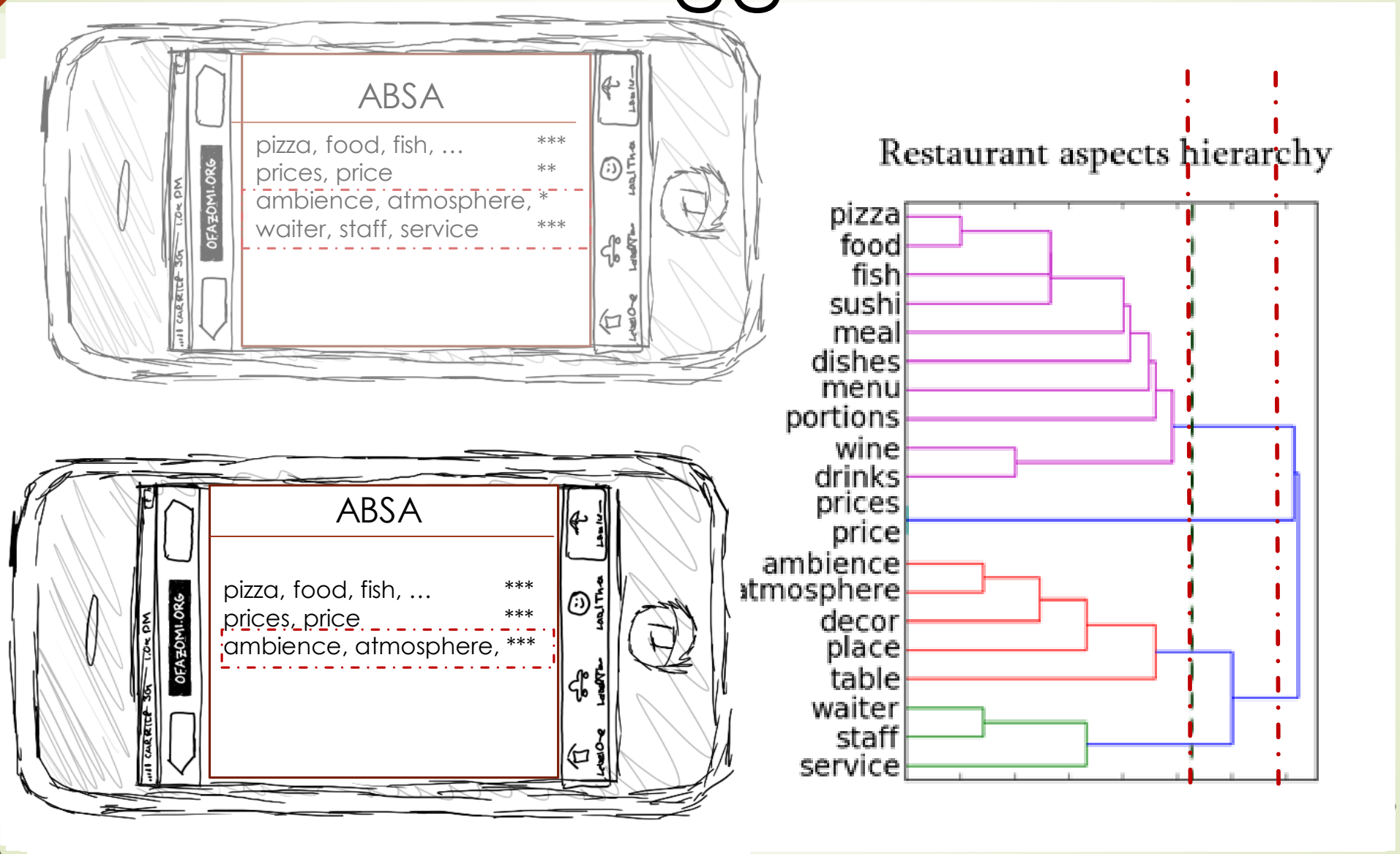
Approaches to aspect aggregation with multiple granularities

- (Near-) synonym grouping (e.g., group "cost" and "price")
  - Only aggregates aspect terms at the lowest granularity
  - E.g., "wine" and "beers" are not synonyms, but they could be aggregated along with "drinks" if a coarser granularity (fewer groups of aspect terms) is desirable
- Predefined Taxonomies
  - Hard to find and to manually construct and maintain
- Flat Clustering aiming at fewer or more clusters of aspect terms
  - E.g., k-means with smaller or larger k
  - Does not satisfy **consistency constraint**: If "wine" and "beers" are grouped together for 5 clusters, they should remain grouped together for 4, 3, and 2 clusters (consistent sense of "zoom out")

- food
- **wine, beers**
- service

- service, **wine**
- food, **beers**

Aspect aggregation via agglomerative clustering



Datasets: agreement problems

Benchmark datasets presuppose *inter-annotator agreement*

- Humans **agree** when asked to **cluster near-synonyms**, but **not** when asked to **produce coarser clusters** of aspect terms
- Humans **don't agree** when **judging given aspect term hierarchies**
- Humans **don't agree** when asked to **create aspect term hierarchies**

But!

- Humans **agree** when asked to fill in an **aspect term similarity matrix**

	food	fish	sushi	dishes	wine
food		4	4	4	2
fish			4	2	1
sushi				3	1
dishes					2
wine					

A two phase decomposition

We propose decomposing aspect aggregation into 2 phases:

- ***Phase A:*** Systems try to produce (fill in) a similarity matrix as close as possible to the gold similarity matrix

- ***Phase B:*** The similarity matrix of *Phase A* is used as a distance measure in hierarchical agglomerative clustering (along with a linkage criterion) to produce an aspect term hierarchy, from which clusterings of different granularities can be obtained.

	food	fish	sushi	dishes	wine
food		4	4	4	2
fish			4	2	1
sushi				3	1
dishes					2
wine					

Phase A data

- **Customer review** data (Restaurants and Laptops)
- **Subjective sentences**, **manually annotated aspect terms**
- **20 most frequently annotated aspect terms** per domain
- **3 human judges** asked to fill in a similarity matrix (1-5)
- **Pearson's rho**: ρ(restaurants) = 0.81, ρ(laptops) = 0.74
- **Absolute agreement**: a(restaurants) = 0.90, a(laptops) = 0.91
- **Gold similarity matrix**: average scores of the 3 judges

	food	fish	sushi	dishes	wine
food		4	4	4	2
fish			4	2	1
sushi				3	1
dishes					2
wine					

Phase A methods

Systems fill in the similarity matrix. The similarity matrix of each system is evaluated by **comparing it to the gold similarity matrix**.

WordNet-based: Wu & Palmer, Lin, Jiang & Conrath, Shortest Path

- No word-sense disambiguation, but greedy approach instead, for aspect terms a_1, a_2: $sim(a_1, a_2)$ = max sense similarity

Distributional (DS): Cosine similarity between $v(a_1)$ and $v(a_2)$

- $v(a) = \langle PMI(a, w_1), \ldots, PMI(a, w_n) \rangle$

AVG: Average of all measures

WN: Average of WordNet-based measures

WNDS: Average of WN and DS

Phase A methods: sense pruning

Sense pruning applied to WordNet-based methods only

- Greedy approach: for aspect terms a_1, a_2 : $\mathrm{sim}(a_1, a_2)$ = max sense similarity
- ***Sense Pruning***: For each aspect term a_i discard some senses s_{ij} before the greedy approach!
- For each sense s_{ij} of aspect term a_i we compute the relevance of s_{ij} to all the other aspect terms a_i'

$$\mathrm{rel}(s_{ij}, a_{i'}) = \max_{s_{i'j'} \in \mathrm{senses}(a_{i'})} \mathrm{sim}(s_{ij}, s_{i'j'})$$

- We take the average relevance of each sense s_{ij} of aspect term a_i to all the other aspect terms a_i'
- For each aspect term a_i we keep its top-5 senses, i.e., the 5 senses with the highest average relevance to the other aspect terms
- The discarded senses are considered to be domain irrelevant

Phase A results: Pearson correlation to gold similarity matrix

	without SP		with SP	
Method	*Rest.*	*Lapt.*	*Rest.*	*Lapt.*
WP	0.475	0.216	0.502	0.265
PATH	**0.524**	0.301	0.529	0.332
LIN@domain	0.390	0.256	0.456	0.343
LIN@Brown	0.434	0.329	0.471	0.391
JCN@domain	0.467	0.348	0.509	0.448
JCN@Brown	0.403	0.469	0.419	0.539
DS	0.283	**0.517**	(0.283)	(0.517)
AVG	0.499	0.352	0.537	0.426
WN	0.490	0.328	0.530	0.395
WNDS	0.523	0.453	**0.545**	**0.546**

A paired *t* test indicates that the differences (with and without pruning) are statistically significant ($p < 0,05$).

Phase A results: Pearson correlation to gold similarity matrix

Mikolov et. al., 2013

Now comparing our best system (WNDS with SP) to two state of the art term similarity methods and human judges

Method	Restaurants	Laptops
Han et al. (2013)	0.450	0.471
Word2Vec	0.434	0.485
WNDS with SP	0.545	0.546
Judge 1	0.913	0.875
Judge 2	0.914	0.894
Judge 3	0.888	0.924

Phase B

- **Get a similarity matrix** (e.g., from a Phase A method or humans)
- **Use the similarity matrix** to compute the **distance between any two aspect terms**
- Choose a **linkage criterion** in effect to compute the **distance between any two clusters** of aspect terms:
  - ***Single***: min distance of any two terms of the clusters
  - ***Complete***: max distance of any two terms of the clusters
  - ***Average***: average distance between the terms of the clusters
  - ***Ward's***: minimum variance criterion (this is not a distance function)
- Use **Hierarchical Agglomerative Clustering** to build an **aspect term hierarchy**
- **Dissect** the aspect term hierarchy at **different depths**, to obtain **fewer or more clusters**.

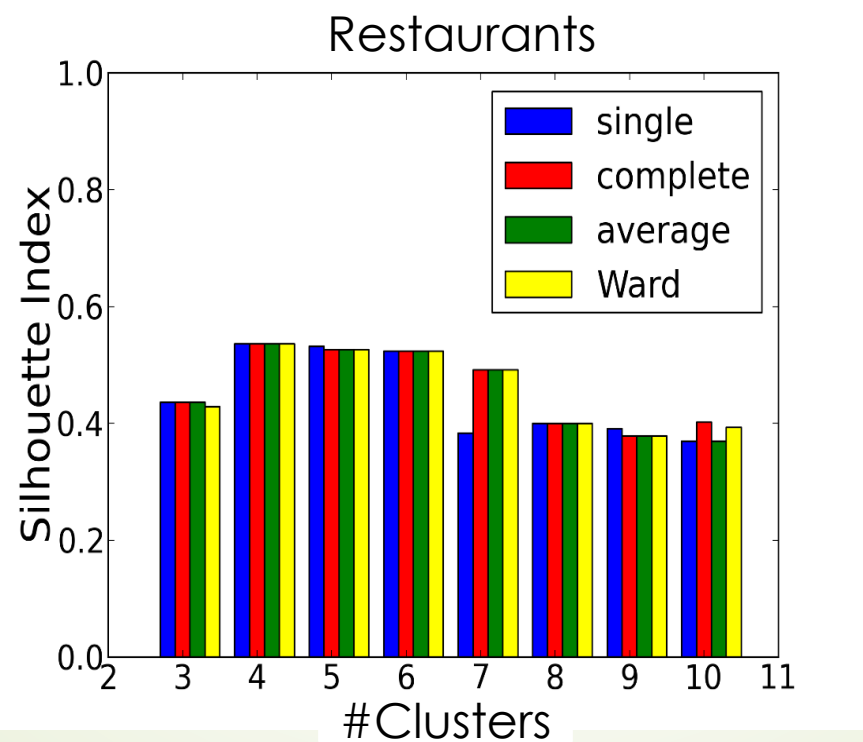
Phase B: evaluation

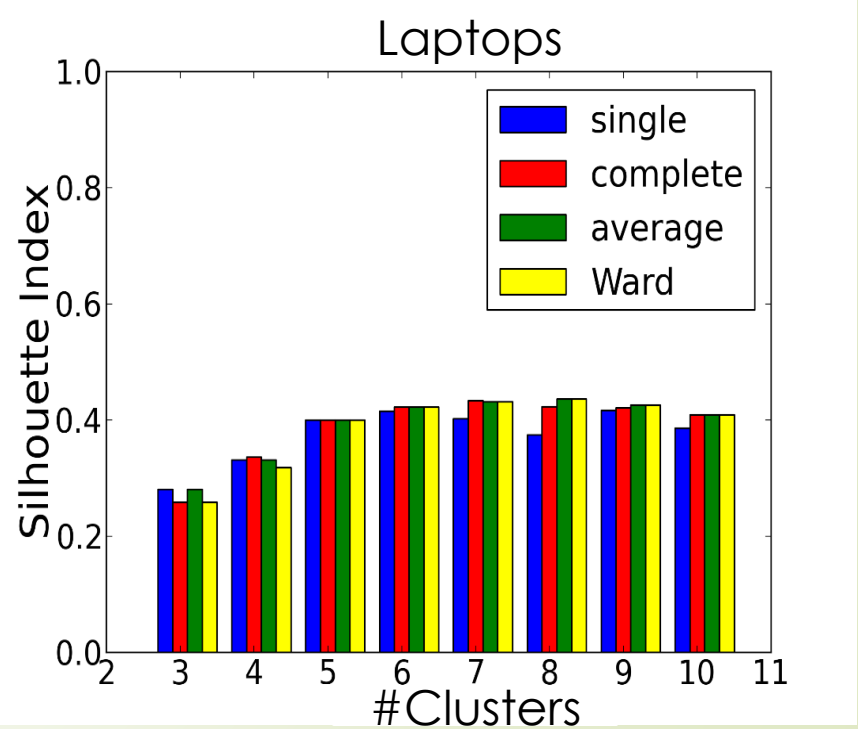
- Silhouette Index (Rousseeuw, 1987)
  - Considers both inter and intra cluster coherence
  - Ranges from -1.0 to 1.0
  - Requires the distances between cluster elements (aspect terms) to be known when evaluating clusters
  - We use the correct distances provided by the gold Phase A similarity matrix
- Different indices produce similar results
  - Dunn Index (Dunn, 1974)
  - Davies-Bouldin Index (Davies and Bouldin, 1979)

Phase B results: gold similarity matrix, different linkage criteria

We use the **gold similarity matrix** from **Phase A** and **Hierarchical Agglomerative Clustering** with **4 different linkage criteria**

- **No linkage criterion** clearly **outperforms** the others
- **All four criteria** perform **reasonably well**

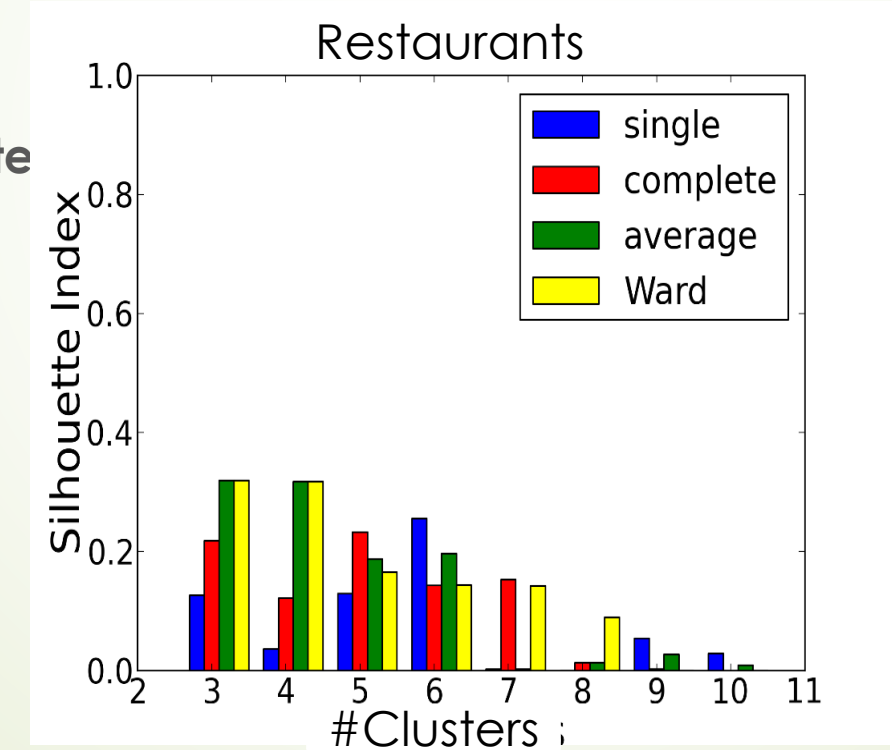


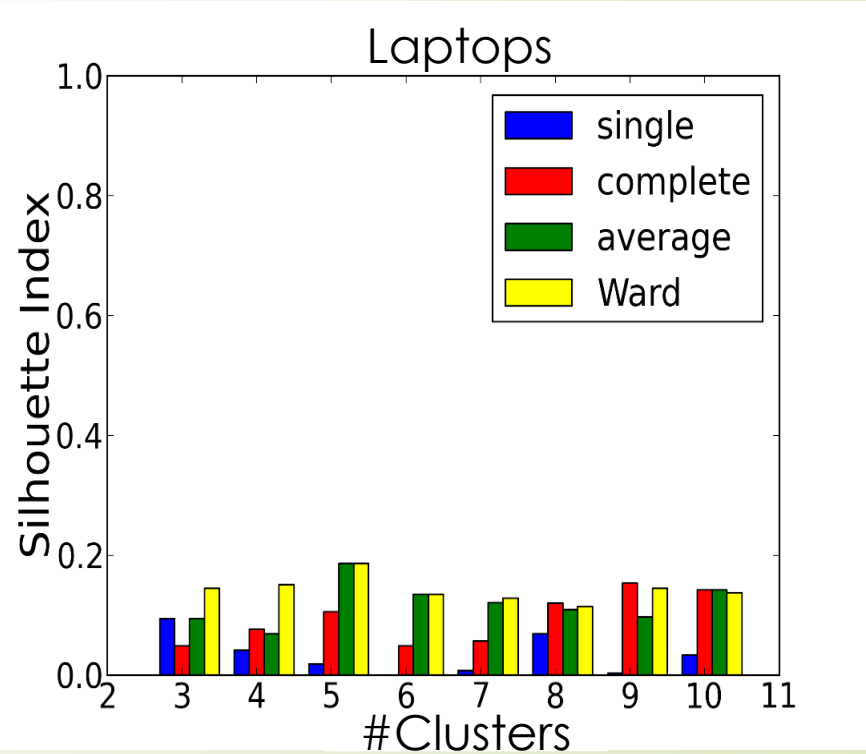


Phase B results: similarity matrix of WNDS+SP, different linkage criteria

We now use the **similarity matrix** of the **best Phase A method** (WNDS+SP) and **Hierarchical Agglomerative Clustering** with **4 linkage criteria**

- Again, **no clear winner** among the **linkage criteria**
- All the **scores deteriorate significantly**

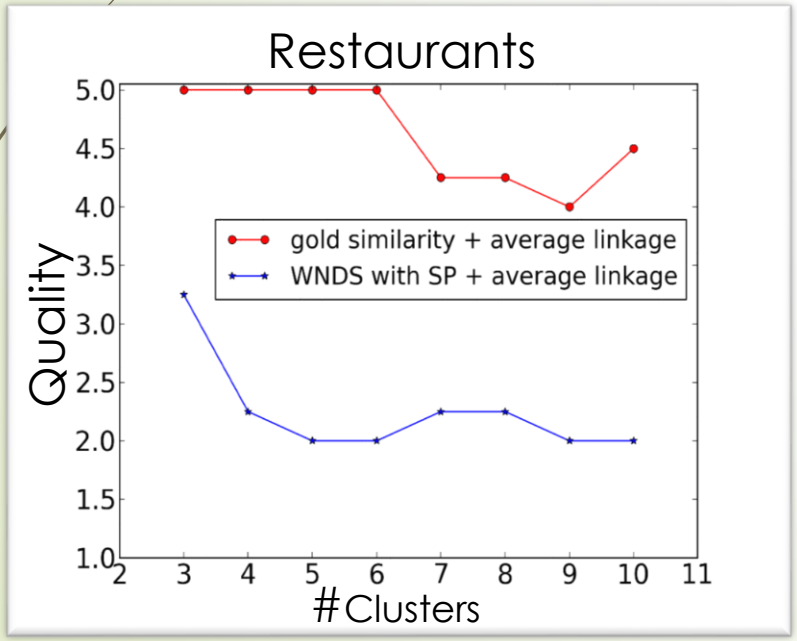


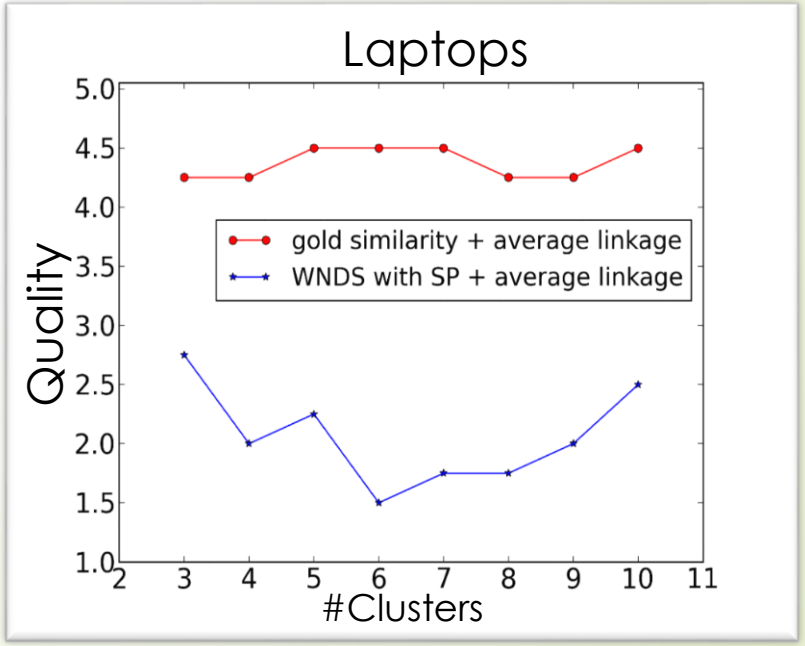


Phase B results: human evaluation

We asked 4 human judges to evaluate (1-5 scale) clusterings of varying granularities (fewer or more clusters)

- **System 1: gold similarity matrix** of Phase A plus **Hierarchical Agglomerative Clustering** (***HAC***) with **average linkage**
- **System2: *WNDS+SP* similarity matrix** plus ***HAC*** with **average linkage**
- **Absolute inter-annotator agreement**: greater than 0.8 in all cases





See our paper for more details (Pavlopoulos and Androutsopoulos, 2014b)

Summary & contribution of section (1/2)

- We introduced **aspect aggregation at multiple granularities** and a **two-phase decomposition**
  - **Phase A fills in** a pairwise **aspect term similarity matrix**
  - **Phase B** uses the **similarity matrix of Phase A**, a **linkage criterion**, and **hierarchical agglomerative clustering** to produce an **aspect hierarchy**
- **Dissecting aspect hierarchy** at **different depths** produces **consistent clusterings at different granularities**
- Our decomposition leads to **high inter-annotator agreement** and allows **previous work** on term similarity and HAC to be reused

See our paper for more details (Pavlopoulos and Androutsopoulos, 2014b)

Summary & contribution of section (2/2)

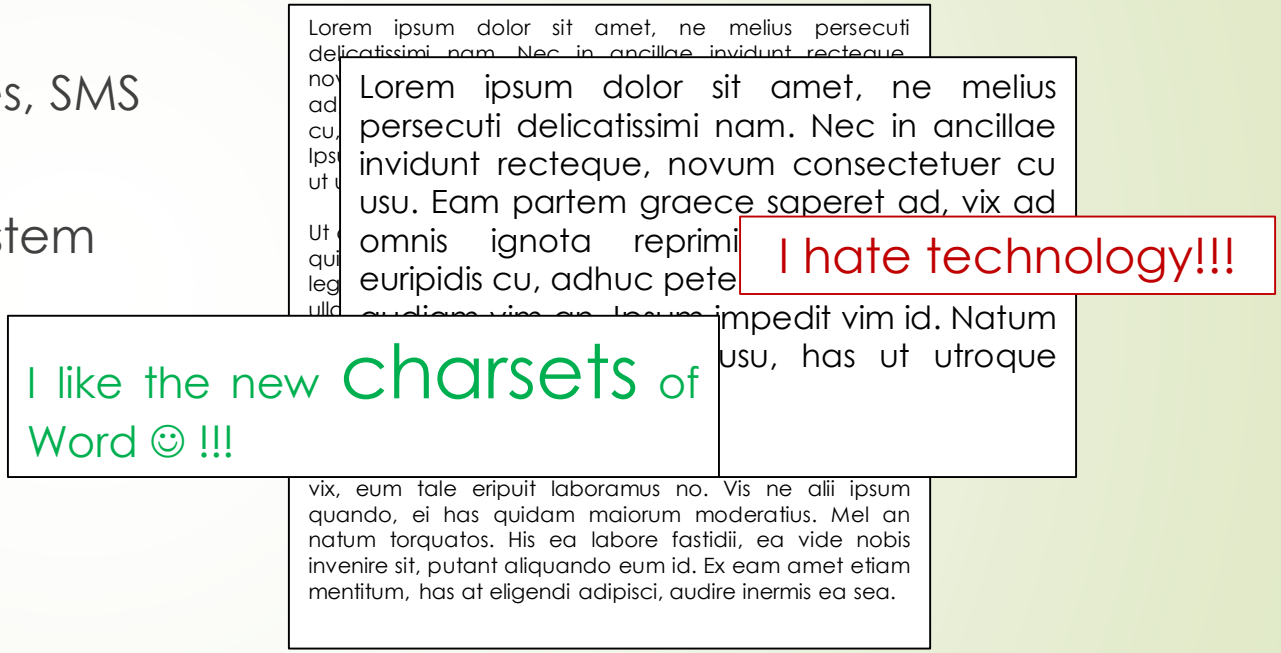
- We introduced a **sense-pruning** mechanism that **improves WordNet-based similarity measures** and leads to the **best performing method** in **Phase A**, but **large scope for improvement**
- With the **gold** Phase A **similarity matrix**, the **quality** (perceived and measured with SI) of the **clusters** of Phase B is **high**, but much **lower quality** with the **similarity matrix** of the **best Phase A method**
- We also provide **publicly available datasets**

Outline

1. Aspect term extraction
2. Multi-granular aspect aggregation
3. Message-level sentiment estimation
4. Aspect term sentiment estimation

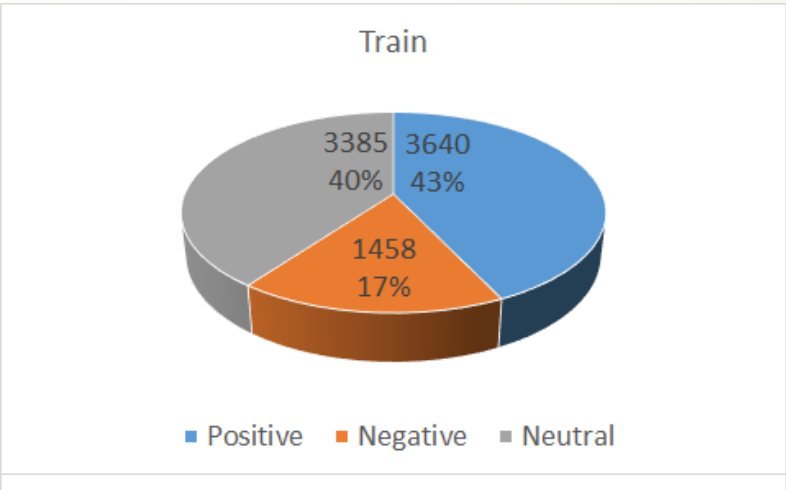
Our message-level sentiment estimation system

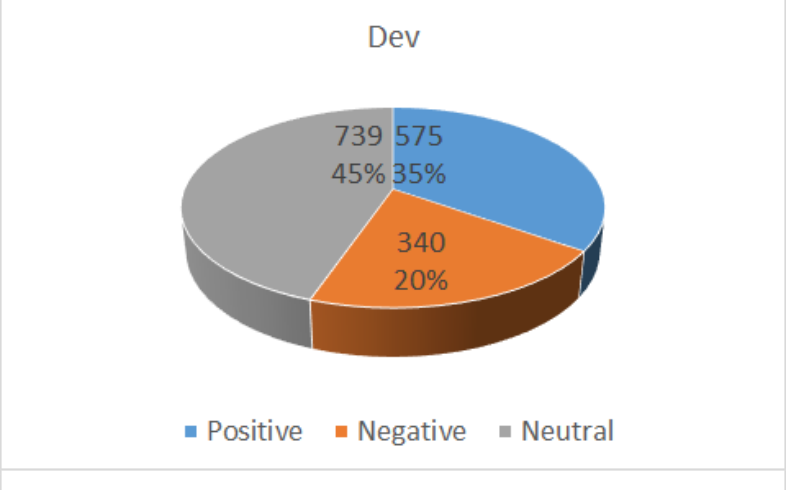
- Messages: sentences, social media updates, SMS
- Our message-level sentiment estimation system
  - Androutsopoulos I.
  - Karampatsis M.
  - Makrynioti N. (2013)
  - Malakasiotis P.
  - Pavlopoulos J.

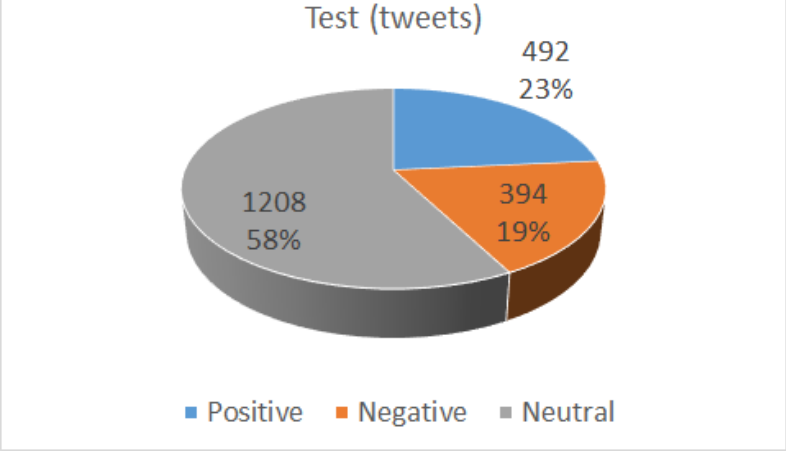


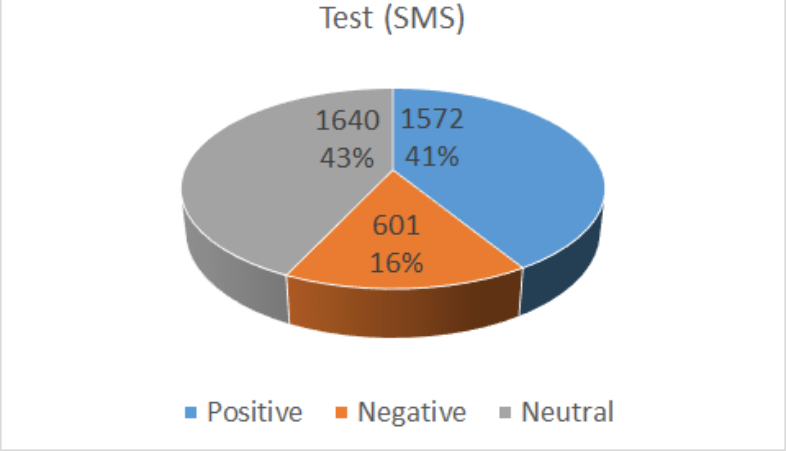
Sentiment analysis In Twitter (2013)

- Goal: Classify each message to positive, negative, or neutral
- Train: 8730 positive, negative, and neutral messages from
  - Originally 9728, but privacy issues…
- Dev: 1654 positive, negative, and neutral messages from Twitter
- Test: 3814 messages from Twitter & 2094 SMS messages

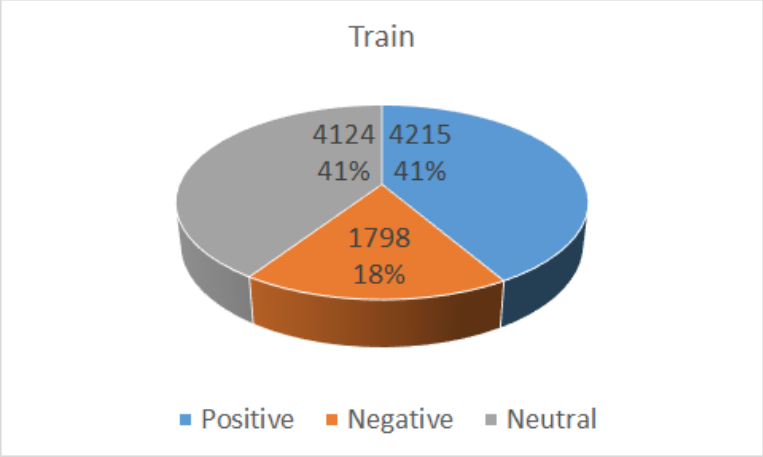


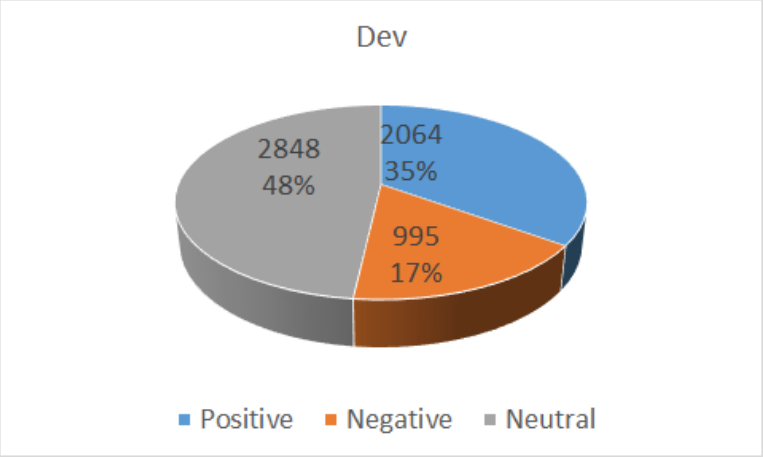






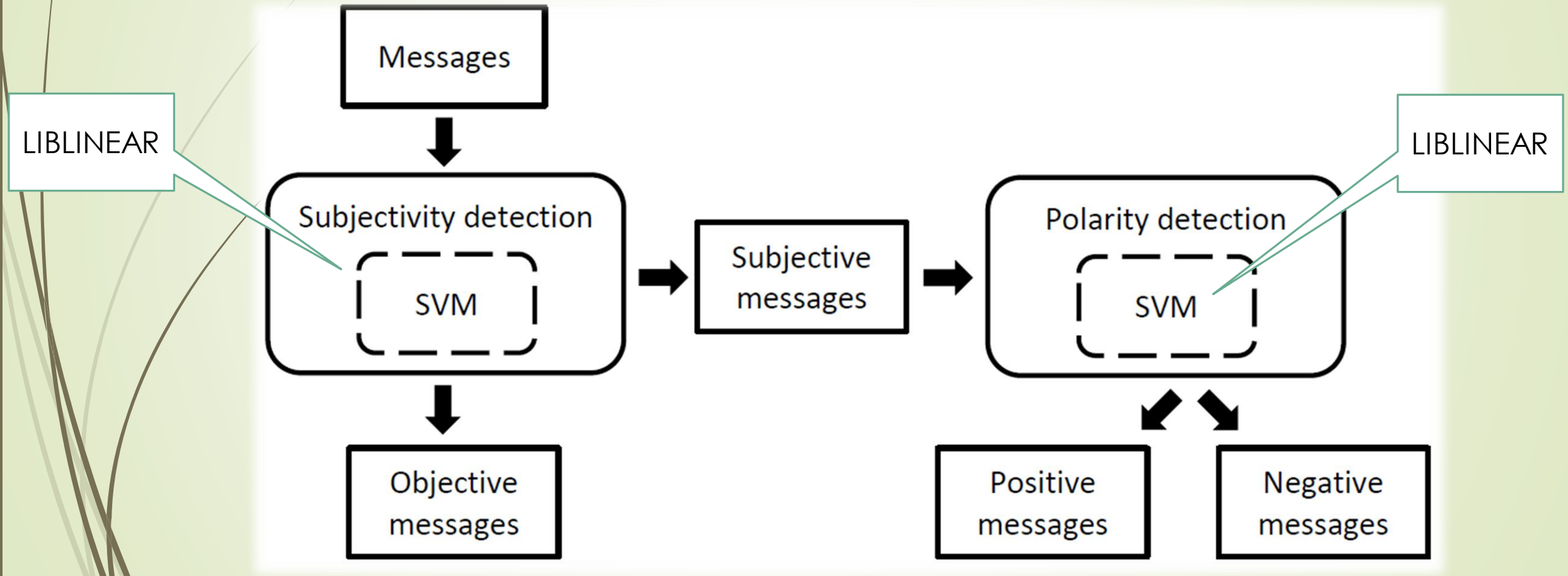
Sentiment analysis In Twitter (2014)





- Goal: Classify each message to positive, negative, or neutral
- Train: 8730 train + 1654 dev messages from Twitter (2013)
- Dev: 3814 + 2094 test messages from Twitter and SMS (2013)
- Test: 8987 tweets, tweets with sarcasm, SMS, messages from blog posts (Live Journal)

Our system's architecture



Data preprocessing

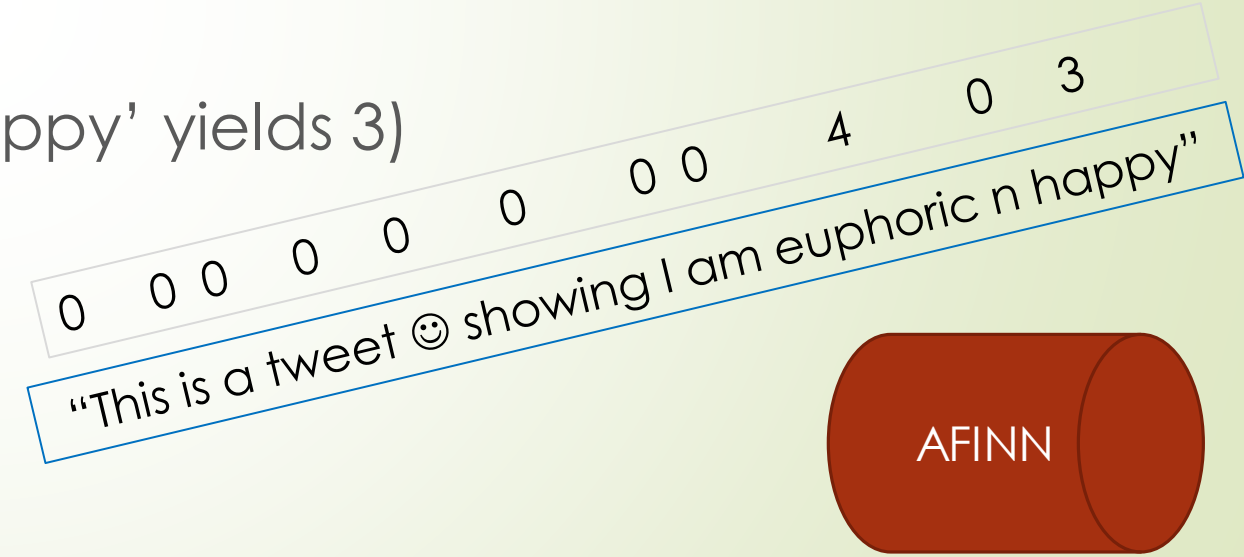
- Twitter-specific tokeniser & POS tagger (Owoputi 2013)
- Text normalization and slang removal
  - edit distance to replace unknown words with their of most similar word in an English dictionary (see also Karampatsis 2012)
  - e.g., flames → angry comments, xmpl → example, etc.

Features

- Morphological (e.g., #elongated_words ‘gooood’, or #capitalized_tokens ‘I WANT MORE’)
- POS-tags based (e.g., #nouns, etc.)
- Sentiment lexicons (e.g., AFINN, SentiWordNet, NRC, MPQA)
  - For lexicons with no scores (e.g., MPQA) we compute our own
- Miscellaneous (e.g., existence of negation, Twitter clusters)

Features: scores of sentiment lexicons

- For each lexicon we compute the following:
  - Sum, max, min, average of scores of the message's words: (7, 4, 3, 3.5)
  - Count of words with scores (2)
  - Score of the last word (e.g., 'happy' yields 3)
  - All features normalized to [0, 1]



Results: Average F1(±)

Test set	Best	AUEB	Median
LJ (2014)	74.84	70.75 (9th/50)	65.48
SMS (2013)	70.28	64.32 (8th/50)	57.53
TW (2013)	72.12	63.92 (21st/50)	62.88
TW (2014)	70.96	66.38 (14th/50)	63.03
TWSarc (2014)	58.16	56.16 (4th/50)	45.77
AVGall	68.78	64.31 (6th/50)	56.56
AVG (2014)	67.62	64.43 (5th/50)	57.97

Summary & contribution of section

- Message-level sentiment estimation system
- 'Sentiment Analysis in Twitter' SemEval task
  - 2013: good rank
  - 2014: better rank
- 2-stage pipeline approach
  - Handles well class imbalance
- Good generalization ability

Work of this section has been published in (Malakasiotis, 2013), (Karampatsis, 2014)

Outline

1. Aspect term extraction
2. Multi-granular aspect aggregation
3. Message-level sentiment estimation
4. Aspect term sentiment estimation

Basis for 'aspect term polarity' ABSA SemEval 2014/2015 task

Task description

The food was delicious!
Nice food but horrible wine and beers.
Excellent service! Thank you ☺
…

ABSA

Aspect term extraction

food, wine, beers, service, …

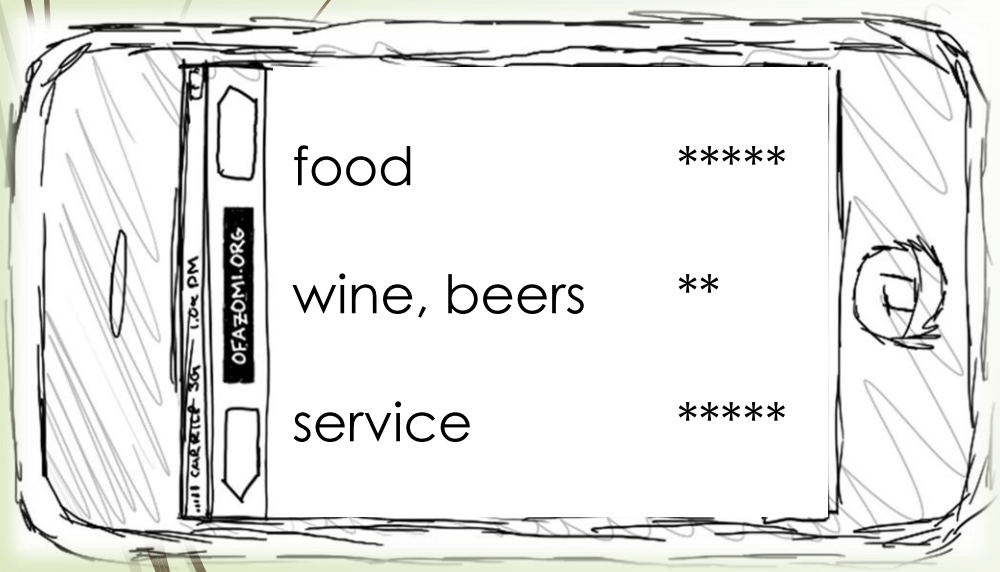
Aspect term aggregation

food, wine, beers, service, …

Aspect term polarity

food, **wine**, **beers**, **service**, …

Visualisation



Aspect term polarity estimation

"Estimate the sentiment polarities of the aspect term occurrences in a sentence"

"I hated their ***fajitas***, but their ***salads*** were great"
"The ***fajitas*** were their starters"
"The ***fajitas*** were great to taste, but not to see"

- Sentiment of the aspect term, not the sentence per se
- Positive, negative, neutral, or conflict
- Subtask of **ABSA** in SemEval 2014

Our aspect term polarity datasets

Domain	**Train**	**Test**	**Total**
Restaurants	3041	800	3841
Laptops	3045	800	3845
Total	6086	1600	7686

Human annotations of aspect term occurrences and their polarities

Inter-annotator agreement: Kappa ≥ 75%

Our waiter was friendly and it is a shame that he didn't have a supportive staff to work with.

Our aspect term polarity datasets

Restaurants	Positive	Negative	Neutral	Conflict	Total
Train	2164	805	633	91	3693
Test	728	196	196	14	1134
Total	2892	1001	829	105	4827

Laptops	Positive	Negative	Neutral	Conflict	Total
Train	987	866	460	45	2358
Test	341	128	169	16	654
Total	1328	994	629	61	3012

Aspect term polarity evaluation: common measures

$$Acc = \frac{|correctly\ classified\ aspect\ term\ occurrences|}{|aspect\ term\ occurrences|}$$

$$Pre_c = \frac{|correctly\ classified\ aspect\ term\ occurrences\ of\ class\ c|}{|aspect\ term\ occurrences\ classified\ as\ c|}$$

$$Rec_c = \frac{|correctly\ classified\ aspect\ term\ occurrences\ of\ class\ c|}{|aspect\ term\ occurrences\ of\ class\ c|}$$

$$F1_c = 2\frac{Pre_c\, Rec_c}{Pre_c + Rec_c},\ where\ c\colon +, -, 0, \pm$$

Aspect term polarity evaluation in ABSA

The **fajitas** were rather good!!!

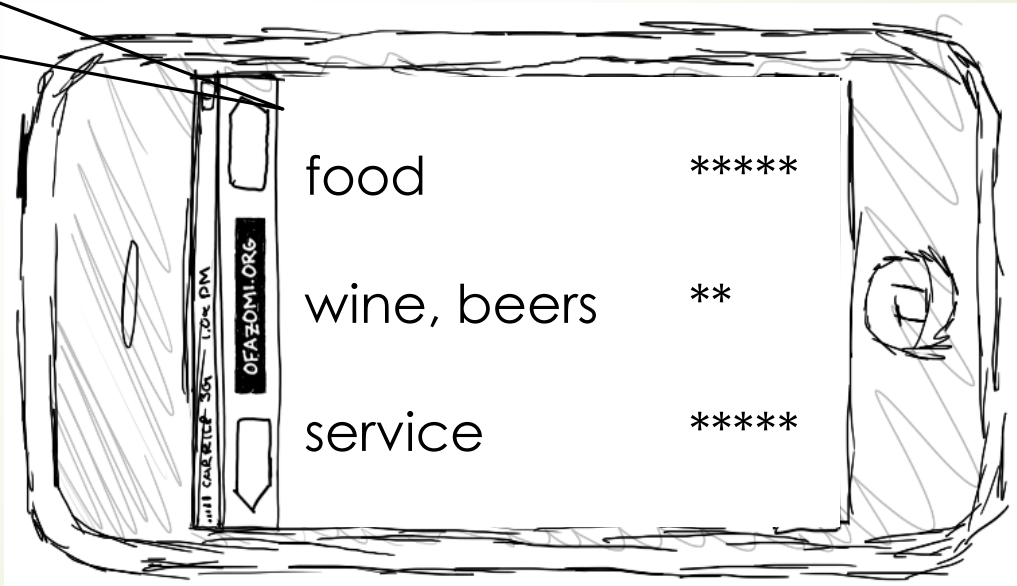
The **fajitas** were rather good!!!

The **fajitas** were rather good!!!

The **fajitas** were rather good!!!

Aspect term polarity evaluation in ABSA

We care more about frequently discussed aspect terms, in ABSA



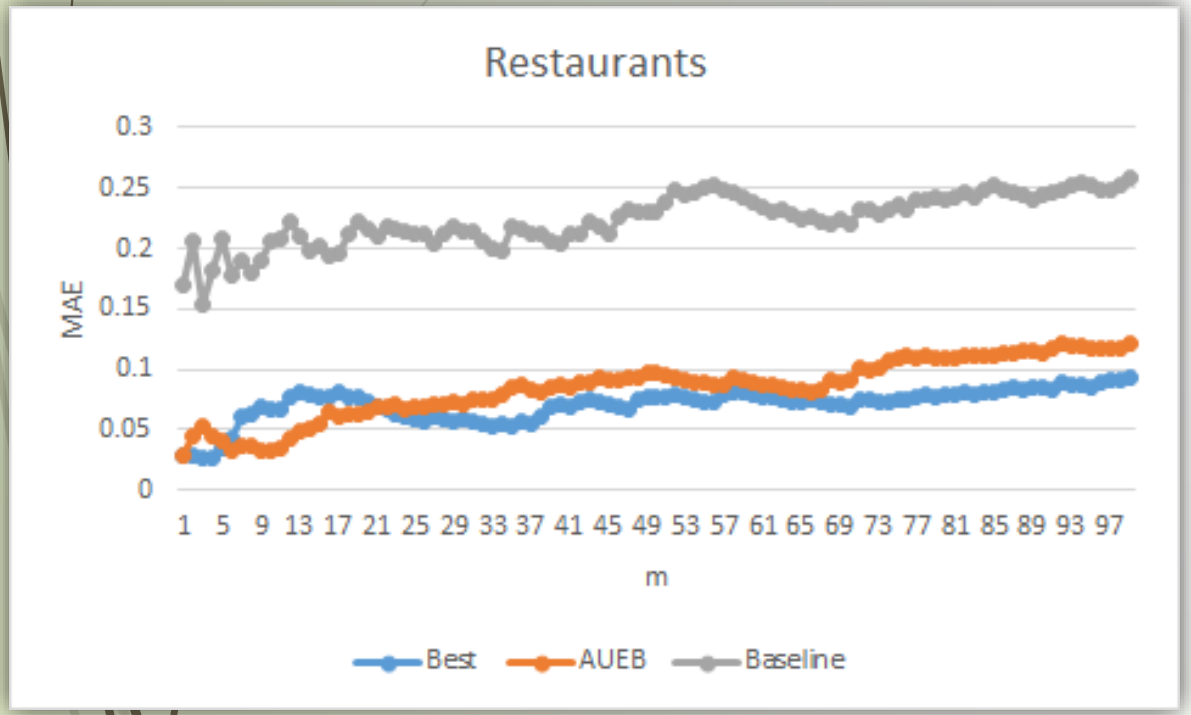
Aspect term polarity evaluation: mean absolute error

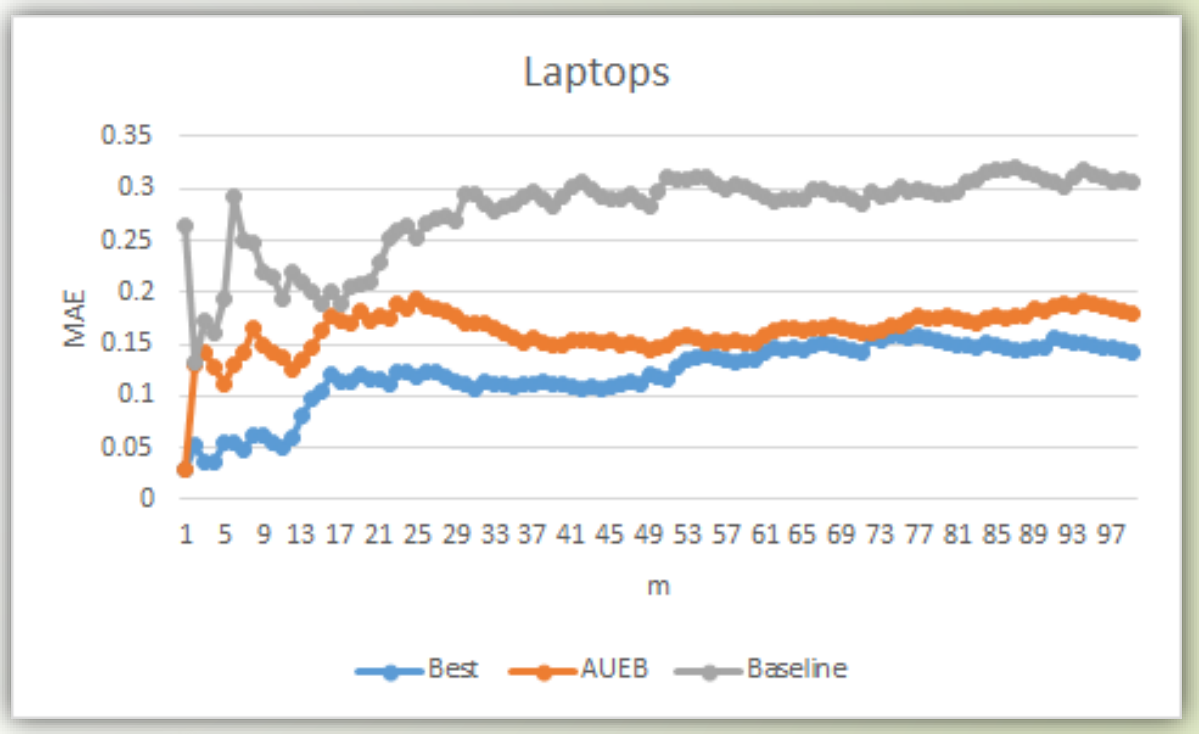
I. For each distinct aspect term a_i we measure its average polarity in all texts
   - E.g., if a_i has 3 positive occurrences, 4 negative and 3 neutral and conflict ones, then: $v_i = \frac{3(+1)+4(-1)+3(0)}{10} = -0.1$
   - We compute both the predicted v_i and the true v_i^* average polarity.

II. Then, for the m **most frequent** (distinct) aspect terms:

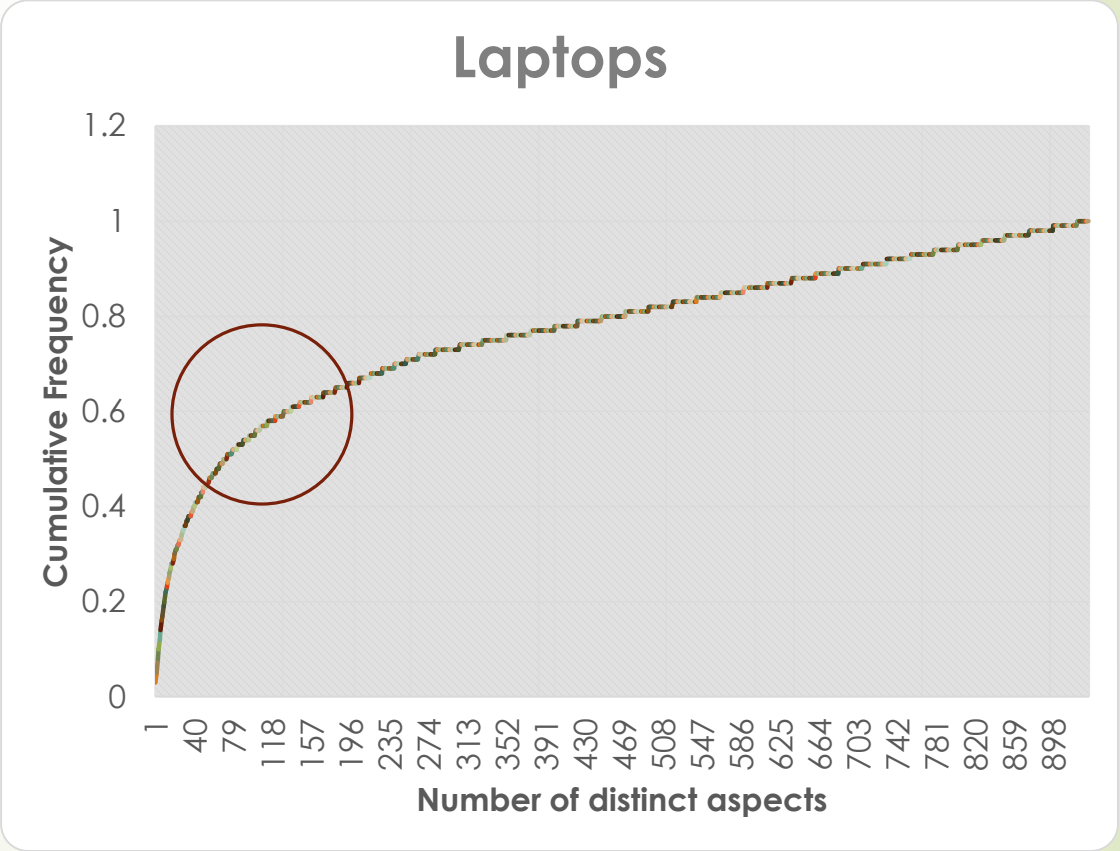
$$MAE_m = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{2}|v_i - v_i^*|$$

Aspect term polarity evaluation: mean absolute error





Aspect term polarity: cumulative frequency



Applying a message-level sentiment estimation system

"I hated their ***fajitas***, and their ***salads***"

Same label for all aspect term occurrences

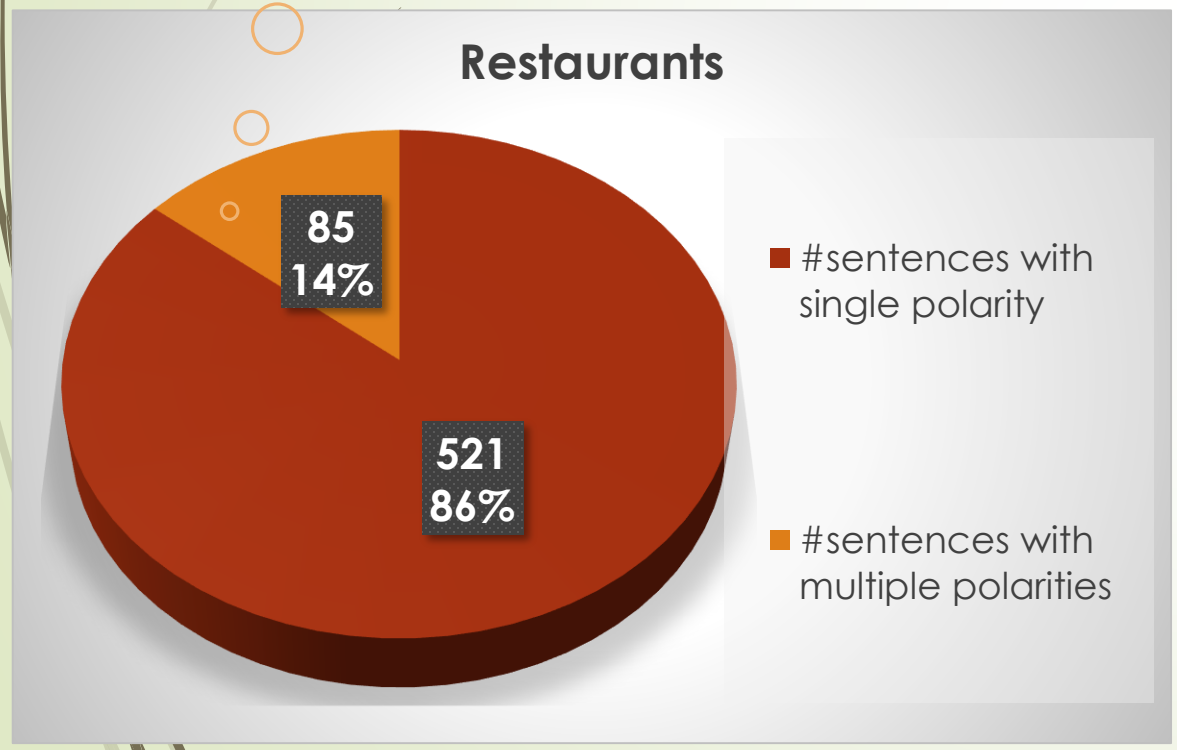
Problem for sentences containing aspect terms with multiple polarities

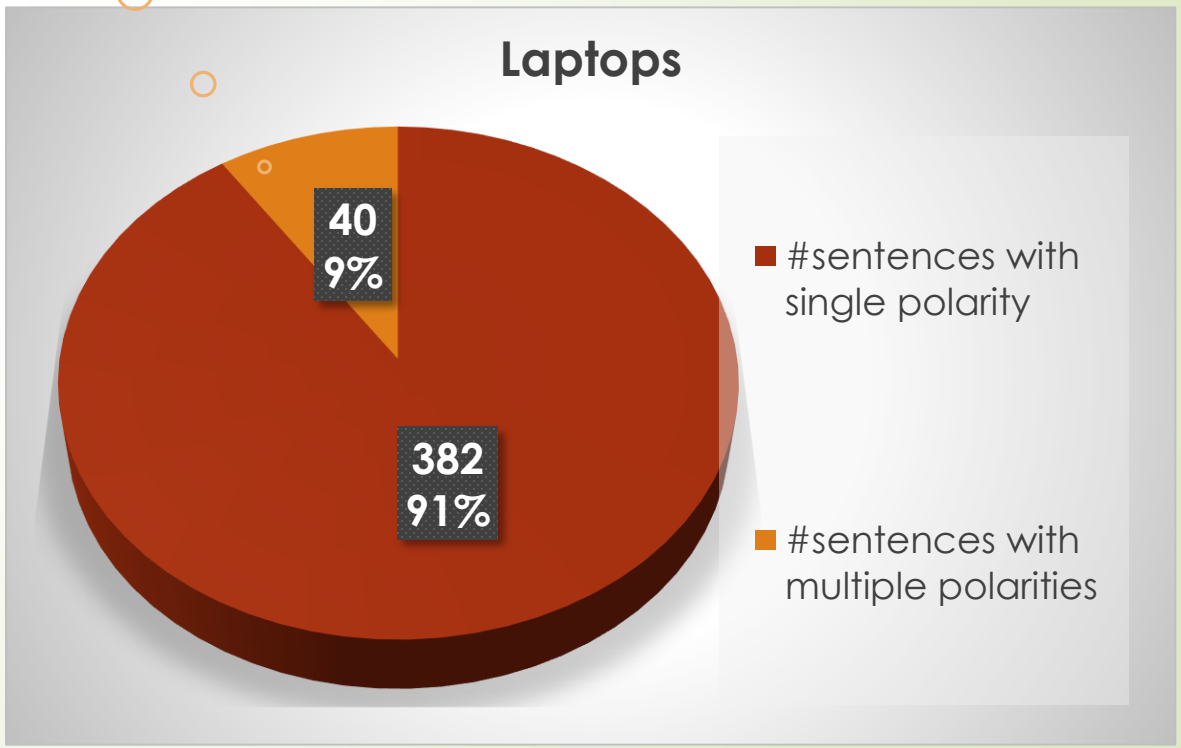
"I hated their ***fajitas***, but their ***salads*** were great"

Multi-polar and single-polar sentences

7-14%
accuracy loss

6-10%
accuracy loss





Evaluation

Teams	Error rate: Restaurants	Error rate: Laptops
DCU	0.191 (1st)	0.295 (1st)
Median	0.292 (14th)	0.414 (14th)
AUEB	0.318 (16th)	0.427 (16th)
Baseline	0.357 (21st)	0.486 (22nd)
Worst	0.583 (24th)	0.635 (23rd)

Evaluation: Restaurants test set

Teams	Error Rate (1-Acc)	$MAE_{m=50}$	$MAE_{m=500}$
DCU	0.191 (1st)/26	0.076 (2nd)/26	0.126 (1st)/26
NRC	0.199 (2nd)/26	0.062 (1st)/26	0.141 (2nd)/26
XRCE	0.223 (3rd)/26	0.088 (4th)/26	0.156 (4th)/26
UWB	0.223 (4th)/26	0.090 (5th)/26	0.143 (3rd)/26
SZTENLP	0.248 (5th)/26	0.120 (10th)/26	0.164 (5th)/26
AUEB	0.318 (16th/26)	0.097 (6th/26)	0.194 (8th/26)
Baseline	0.357 (22nd/26)	0.769 (26th/26)	0.737 (24th/26)

Teams with multiple submissions are shown under one name

Evaluation: Laptops test set

Teams	Error Rate (1-Acc)	$MAE_{m=50}$	$MAE_{m=500}$
DCU	0.295 (1st/26)	0.118 (3rd/26)	0.165 (3rd/26)
NRC	0.295 (2nd/26)	0.141 (12th/26)	0.160 (2nd/26)
IITPatan	0.330 (3rd/26)	0.111 (1st/26)	0.178 (4th/26)
SZTENLP	0.330 (4th/26)	0.124 (4th/26)	0.190 (7th/26)
UWB	0.333 (5th/26)	0.118 (2nd/26)	0.182 (5th/26)
AUEB	0.427 (16th/26)	0.147 (13th/26)	0.201 (11th/26)
Baseline	0.486 (25th/26)	0.704 (25th/26)	0.687 (25th/26)

Evaluation: with an ensemble (restaurants)

Same as above, but instead of AUEB, the system uses UWB

Ensembles	Error Rate	$MAE_{m=50}$
EC1-AUEB	0.202	0.070
EC2-AUEB	0.196	**0.058★**
EC1-UWB	0.198	0.118
EC2-UWB	**0.184**	0.075
Best	0.190	0.076

Evaluation: with an ensemble (laptops)

Ensembles	Error Rate	$MAE_{m=50}$
EC1-AUEB	**0.269★**	0.114
EC2-AUEB	0.282	0.108
EC1-UWB	0.283	**0.100**
EC2-UWB	0.282	0.120
Best	0.295	0.118

Summary & contribution of this section

- New datasets
- New evaluation measure
- Message-level sentiment estimation system applied to aspect term sentiment estimation
  - Good performance on the aspect term polarity task, especially **with MAE** or when integrated **in an ensemble**
- The 'Aspect term polarity' subtask of ABSA SemEval 2014 & 2015 was based on the work of this section

Work of this section will appear in (Pontiki et al. 2014)

Contributions of this thesis (1/4)

- Clear decomposition of Aspect Based Sentiment Analysis (ABSA)
  - Systems may compare to each other
- ABSA SemEval task (2014, 2015) based on the work of this thesis

Contributions of this thesis (2/4)

- Introduced 3 new aspect term extraction datasets
  - Laptops/Restaurants/Hotels
  - Showed that domain variety is important
- New aspect term extraction evaluation measures
  - Weighted precision, weighted recall, average weighted precision
- The 'Aspect term extraction' subtask of ABSA SemEval 2014 & 2015 was based on the work of this section

Contributions of this thesis (3/4)

- Introduction of a Multi-granular Aspect Aggregation ABSA step
- Two-phase methodology for Multi-granular Aspect Aggregation
- Sense pruning mechanism which improves WordNet-based measures and leads to best performing method
- Publicly available datasets

Contributions of this thesis (4/4)

- 2-stage sentiment estimation system
  - Good generalization ability
  - High rank in Sentiment tasks of SemEval 13/14
- Ensemble of classifiers
  - Best results in 'Aspect Polarity' subtask of ABSA task in SemEval '14
- Mean Absolute Error evaluation measure

Publications

J. Pavlopoulos and I. Androutsopoulos, "Aspect Term Extraction for Sentiment Analysis: New Datasets, New Evaluation Measures and an Improved Unsupervised Method". 5th Workshop on Language Analysis for Social Media (LASM 2014), Proc. of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, pp. 44-52, 2014.

J. Pavlopoulos and I. Androutsopoulos, "Multi-Granular Aspect Aggregation in Aspect-Based Sentiment Analysis". Proc. of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, pp. 78-87, 2014.

M. Pontiki, D. Galanis, **J. Pavlopoulos**, H. Papageorgiou, I. Androutsopoulos, S. Manadhar, "SemEval-2014 Task 4: Aspect Based Sentiment Analysis". Proc. of the 8th International Workshop on Semantic Evaluation . Dublin, Ireland, 2014. (to appear)

P. Malakasiotis, R.-M. Karampatsis, N. Makrynioti, and **J. Pavlopoulos**, "nlp.cs.aueb.gr: Two Stage Sentiment Analysis". Proc. of 7th International Workshop on Semantic Evaluation, Atlanta, Georgia, U.S.A, 2013.

R.-M. Karampatsis , **J. Pavlopoulos**, and P. Malakasiotis, " AUEB: Two Stage Sentiment Analysis of Social Network Messages". Proc. of 8th International Workshop on Semantic Evaluation, Dublin, Ireland, 2014. (to appear)

P. Alexopoulos and **J. Pavlopoulos**, "A Vague Sense Classifier for Detecting Vague Definitions in Ontologies". Proceedings of EACL, Gothenburg, Sweden, pp. 33-37 (short papers), 2014

P. Alexopoulos, **J. Pavlopoulos** and Ph. Mylonas, "Learning Vague Knowledge From Socially Generated Content in an Enterprise Framework". Proceedings of the 1st Mining Humanistic Data Workshop, Halkidiki, Greece, 2012.

G. Anadiotis, K. Kafentzis, **J. Pavlopoulos** and A. Westerski, "Building Consensus via a Semantic Web Collaborative Space". Proceedings of the Semantic Web Collaborative Spaces Workshop, (WWW 2012), Lyon, France, pp. 1097-1106, 2012.

P. Alexopoulos, **J. Pavlopoulos**, M. Wallace, and K. Kafentzis, "Exploiting ontological relations for automatic semantic tag recommendation". Proceedings of I-Semantics, New York, NY, USA, pp. 105-110, 2011.

Thank you!

Questions?