



Architectures of Meaning, A Systematic Corpus Analysis of NLP Systems

[Link to publication record in Manchester Research Explorer](#)

Citation for published version (APA):

Wysocki, O., Florea, M., Landers, D., & Freitas, A. (2021). Architectures of Meaning, A Systematic Corpus Analysis of NLP Systems. *ArXiv*.

Published in:

ArXiv

Citing this paper

Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights

Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy

If you believe that this document breaches copyright please refer to the University of Manchester's Takedown Procedures [<http://man.ac.uk/04Y6Bo>] or contact uml.scholarlycommunications@manchester.ac.uk providing relevant details, so we can investigate your claim.



ARCHITECTURES OF MEANING

A SYSTEMATIC CORPUS ANALYSIS OF NLP SYSTEMS

Oskar Wysocki^{1,2}, Malina Florea¹, Dónal Landers², and André Freitas^{1,2,3}

¹Department of Computer Science, The University of Manchester
²digital Experimental Cancer Medicine Team, Cancer Biomarker Centre,
CRUK Manchester Institute, University of Manchester
³Idiap Research Institute

ABSTRACT

This paper proposes a novel statistical corpus analysis framework targeted towards the interpretation of Natural Language Processing (NLP) architectural patterns at scale. The proposed approach combines saturation-based lexicon construction, statistical corpus analysis methods and graph collocations to induce a synthesis representation of NLP architectural patterns from corpora. The framework is validated in the full corpus of Semeval tasks and demonstrated coherent architectural patterns which can be used to answer architectural questions on a data-driven fashion, providing a systematic mechanism to interpret a largely dynamic and exponentially growing field.

1 Introduction

Natural Language Processing (NLP) systems have been subjected to a Cambrian explosion of architectural paradigms in the past few years. The scale on the number of contributions and its exponential growth, bring challenges in understanding how NLP architectural patterns evolve and consolidate in different sub-areas and tasks.

This paper aims to provide the methodological support for the interpretation of NLP architectural patterns at scale by applying statistical corpus analysis methods over large-scale NLP corpora. We analyse the use of corpus statistics to compute large-scale collocation patterns jointly with graph visualisation methods as a device to interpret architectural patterns at scale. The proposed methods aims to address questions such as:

- What is the complete list of architectural patterns present in NLP?
- What are the prevailing architectural patterns (classifiers, layers, regularisation, linguistic resources) for each NLP task?
- How these patterns are evolving over time and what are the emerging consolidated/canonical architectural motifs?
- What is the architectural variance across different subareas?

This paper proposes these specific novel contributions:

1. The use of a systematic statistical corpus analysis to interpret architectural features of NLP systems.
2. The extension of existing corpus statistics methods with a-priori sequence mining to support the observation of architectural motifs.
3. The proposal of collocation graphs to support the interpretation of more complex architectural motifs.
4. The validation of the proposed model by the observation of the evolution and consolidation of distinctive architectural patterns.
5. The publication of a framework for systematic corpus analysis of architectural patterns which can be transported to other AI/NLP corpora.

This paper is organised as follows: Section 2 provides a critical analysis of existing corpus statistics and adapt them to support the proposed architecture mining framework; Section 3 describes the creation of the target corpus, which is followed by Section 4 which describes the construction of the supporting component lexicon. Section 5 provides an empirical analysis of the proposed method for interpreting NLP architectural patterns, which is followed by related work and conclusions.

2 Statistical Corpus Analysis

In this section we introduce the corpus statistics measures targeting the extraction of architectural patterns.

2.1 Relative frequencies

The first step in the analysis was to investigate the relative frequency of a single component: $freq_{rel} = \text{frequency in given year} / \# \text{ of papers in given year}$. Both single and multiple appearances of a component mention in a paper count as a binary feature. The advantage of using *relative frequency* instead of *frequency* is to account for a variation in topic popularity along different years (Fig.1).

To analyse trends over the years, for each $freq_{rel}$ data point we fit a linear regression model (LR). We used the LR coefficient a , which signifies the slope of the line, and r^2 to measure of how well observed values are replicated by the model (Fig.2).

2.2 Collocations

A *collocation* is defined as a co-occurrence of two or more components in the list of components found in a targeted section of a paper. In order to discover emerging, fading or prevailing collocations over the years, first we propose a pairwise component collocation measure using G^2 (the likelihood ratio test statistics), extending it to longer collocations of up to 8 components using the Apriori algorithm [1].

In order to define the collocation measure, a co-occurrence matrix was computed for each year, where row names and column names consist of components, and values indicate how many papers two components co-occurred. The most frequent components are more likely to be part of frequent collocations. Thus, in order not to overlook less frequent but significant collocations, we considered 4 statistical measures: *Point-wise Mutual Information* (PMI), *t-score*, likelihood ratio test statistics G^2 and *odds ratio* (α). PMI tends to strongly favor low-frequency words, whereas *t-score* does just the opposite. G^2 appears to be a quasi-standard and was used to rank collocations [7]. Although G^2 is used here as a measure of significance, another measure i.e. α must be used to evaluate whether occurrence of components is

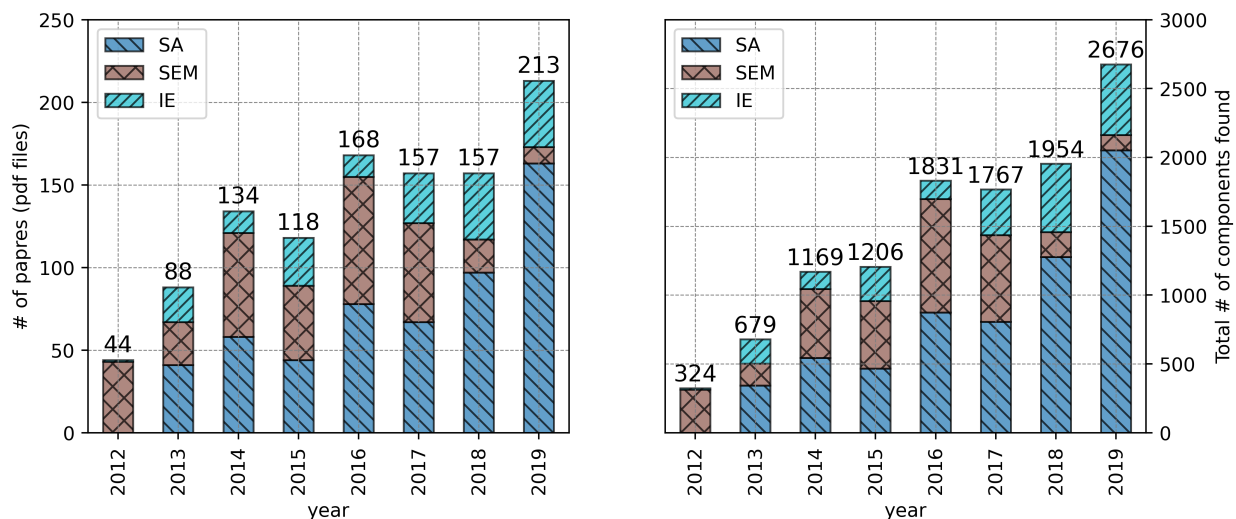


Figure 1: # of papers with a NLP system description submitted to SemEval (left) and the total # of components found in these papers in years 2012-2019 (right).

positively or negatively associated. α ranges from 0 to ∞ , where values >1 and <1 indicate positive and negative association respectively. Collocations with $\alpha < 1$ were excluded from the ranking.

To investigate collocations of 3 or more words, we used the Apriori algorithm, which performs frequent item set mining and association rule learning. A rule is defined as $X \Rightarrow Y$, where X and Y are left and right hand side part of the rule respectively. Both X and Y can be a single component or a set of components. Rules were learned using the *Efficient-Apriori* Python implementation ([13]), with the parameters set on $supp_{min} = 0.1$, $conf_{min} = 0.3$. This parametrisation expresses that only the components present in at least 10% of papers in a given year, and that only the rules appearing in at least 30% of lists containing the left-hand side rule component are considered. There is a variety of measures for the evaluation of association rules [14]. We decided to rank the rules using G^2 , both due to its statistical significance estimation power and for consistency with 2-components collocations comparison. Again, *odds ratio* was used to remove negatively association from the ranking.

$$conf(X \Rightarrow Y) = \frac{supp(X \cap Y)}{supp(X)}$$

$$supp(X) = \frac{|t \in T; X \subset t|}{|T|}$$

where T - set of lists, t - list of components, X - set of components. For the sake of space, we decided to present the top 10 rules in a given year of SemEval. However components tend to be repeated in rules with similar significance (e.g. $(A, C, E) \Rightarrow (B, D)$ and $(A, C) \Rightarrow (B, D, E)$). Thus, as the main aim is to draw attention to components rather than to precise rules, each rule is converted into a collocation by merging the components from the X and Y (e.g. $(A, C, E) \Rightarrow (B, D)$ into (A, B, C, D, E)). Duplicated collocations are removed, keeping the highest G^2 . G^2 and α are given by:

$$G^2 = 2 * \sum_{i=1}^4 obs * \log \frac{obs}{exp} = 2 * (a_{obs} \log \frac{a_{obs}}{a_{exp}} + b_{obs} \log \frac{b_{obs}}{b_{exp}} + c_{obs} \log \frac{c_{obs}}{c_{exp}} + d_{obs} \log \frac{d_{obs}}{d_{exp}})$$

$$\alpha = \frac{a \times d}{b \times c}$$

where a, b, c and d correspond to the frequencies of collocations $A\&B$, $\bar{A}\&B$, $A\&\bar{B}$ and $\bar{A}\&\bar{B}$ (for 2-components collocations) or $X \cap Y$, $\bar{X} \cap Y$, $X \cap \bar{Y}$ and $\bar{X} \cap \bar{Y}$ (for Apriori rules), *obs* - observed, *exp* - expected. a, b, c and d are defined by the contingency table (Tab.1), which is also used to compute expected values.

2.3 Collocation graphs

To compare results of top frequencies, collocations and their changes over the years, we used *network graphs* [9] as a joined collocation and visualisation device. Graphs were plotted using *Gephi* [2]. Layouts were first computed using the *ForceAtlas 2* algorithm, and subsequent *Noverlap* (to prevent overlapping nodes) and label adjustment were applied. Each node in a graph corresponds to a component, and each edge to a collocation. Collocation's G^2 was used as a weight in *ForceAtlas 2*. Only the top 100 (ranked by G^2) collocations were included. It resulted in a constant n# number of edges, but a varying # of nodes, which additionally informs about diversity in a given sample.

Table 1: Contingency table used for the calculation of G^2 and α for a collocation $A\&B$. A and B are components.

	A	\bar{A}	
B	a	b	a+b
\bar{B}	c	d	c+d
Totals	a+c	b+d	a+b+c+d

3 Creating the Target Corpus

In order to automatically and systematically analyse NLP systems (in terms of components, representations and features) we curated 2 Corpora, named **SemEval Corpus** and **ACL Corpus**. Both are based on scientific research publications from a top NLP conference and the major task-based challenge workshop.

SemEval Corpus: All papers from the SemEval Proceedings from 2012 to 2019 were collected, corresponding to 1341 papers. Each paper was converted into a set of text blocks (using PyMuPDF), which were further mapped into labeled sections (e.g. Abstract, Introduction, References). Data cleaning techniques used for each publication included: i) used a rule-based method to avoid repetitions, short blocks and other data modalities, including page numbers or values from tables; ii) removed citations (e.g. 'Johnson et.al 2015'), punctuation marks, numbers (excluding those in names, e.g. w2v) and stop-words; iii) Keeping all paragraphs, removing the "References" sections; iv) the resulting text was tokenised and converted into a list of sentences. The SemEval Corpus consists of a set of 152000 sentences.

ACL Corpus: All papers from ACL Proceedings from 2010 to 2019 were collected, corresponding to 2265 papers files in total. Using the same steps as for SemEval Corpus, we obtained the ACL Corpus, which was a set of 480000 sentences. Both corpora are available at the project repository (anonymised link). The Semeval corpus will be used to evaluate the proposed approach, while the ACL corpus is used to build the component extractor.

4 Target terms & Component lexicon

A critical component of the proposed framework is the creation of a comprehensive lexicon which describes the architectural components including types of classifiers, feature extraction methods, feature representations, gold-standards, data pre-processing methods, among others.

We developed a *saturation-based method* which coordinated iterative cycles of manual and automatic extraction of components from NLP literature until a plateau in the number of new lexical items was observed (stop point). This method used the iterative application of three steps:

Seed lexicon: Building a seed lexicon by the manual curation 2 Machine Learning technical textbooks and 6 comprehensive surveys focused on Sentiment Analysis and Question Answering systems (references described in the Appendix). This list was later enriched with the support of a senior domain expert in NLP. All terms were manually extracted and normalised to a canonical form, adding 322 terms to the lexicon.

Component classifier: Construction of a Naive-bayes classifier using morphological-level features and context words, in order to determine whether a token is part of a component name or not. The classifier was built over the ACL corpus. In the first cycle the classifier reported 3000 new distinct terms, where 192 of them proved to be true new positive cases.

Gold standard: Manually annotated over 100 randomly sampled papers from the ACL corpus (annotated component mentions). This supported the evaluation of the performance of the algorithm for automatic component extraction (described later in the text) and a post-hoc update of the lexicon with 5 additional items.

The final lexicon consists of 151 unique canonical components (entities), which are represented by one or more lexical expressions entities, e.g. CNN: cnn, convnet, convolutional neural net, among others, resulting in the total of 519 lexical items. The final lexicon is available in the Appendix.

Components extraction We clustered all SemEval papers from 2012 to 2019 tasks into 6 macro-categories: Sentiment Analysis (SA), Semantic Analysis (SEM), Information Extraction (IE), Question Answering (QA), Machine Translation (MT) and Other (OT). 74 (77%) tasks out of the total 96 are tasks SA, SEM or IE: 21, 34 and 19 respectively (summary in the Appendix). 92% (1750/1893) of participating teams and 82% (1079/1314) of submitted papers (with a description of competing system) are associated with SA, SEM and IE tasks. Thus, further analysis includes only SA, SEM and IE related papers, considered as the core of SemEval campaigns. We collected all contributions papers (excluding task description papers).

From each file, text content was extracted using the PyMuPDF python package. Only the sections related to the description of approaches and systems were kept, filtering by a lexicon of section headings. Text was cleaned and normalised using heuristic rules. Both the headings and the preprocessing rules are fully described in the Appendix. The lexicon maps different lexical forms to a single canonical entity and for a single paper the presence of a component defines a binary features (whether the component is present or not).

5 Results and discussion

5.1 Lexicon evaluation

In order to evaluate our algorithm for automatic component extraction, we developed a Gold Standard for the extraction of NLP architectural components (consisting of 979 examples). It consists of a list of components, one for target sampled paper (out of 100). The evaluation of the extraction method reached a recall=0.884 (std=0.206) and a precision=0.670 (0.227).

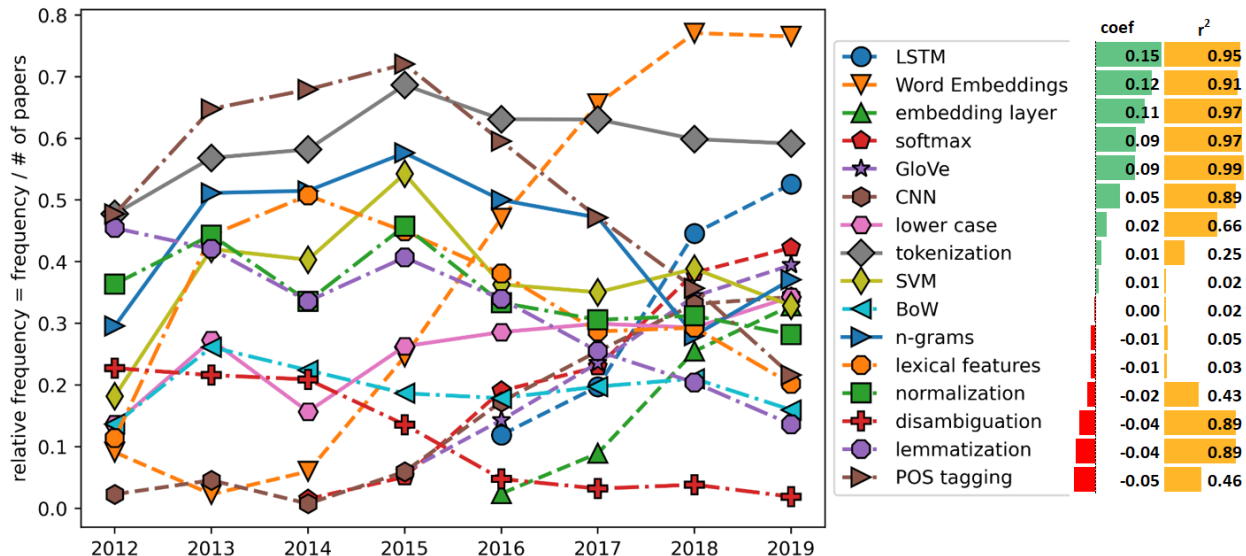


Figure 2: The most frequent components in 2013 (dashdot line), in 2019 (dashed line), or both in 2013 and 2019 (solid line). *Coef* is the slope of the regression model for each component and signifies increase or decrease in frequency over years.

5.2 Statistics for SemEval

There is an increasing trend in the number of papers submitted to SemEval, which correlates with higher total # of components found in these papers in a given year (Fig.1). Note, that the proportion of papers associated with SA, SEM and IE changes, as well as proportion of # of components. SA gains popularity as SEM decreases, what may be one of the factor in evaluating the shift from one group of components to another over the years.

Over the years we observe an increase in the average number of components found in a paper (Fig.3a), which may have several causes. First, novel systems simply are built in a more sophisticated manner which can entail more components. Researchers experiment more with the architecture and use different components for specific subtasks, taking advantage of variety of available methods developed over the years. Second, the lexicon may be biased in terms of number of unique components towards novel algorithms. For example, novel systems based on BiLSTM are likely to be collocated with at least several other components (related to neural networks, e.g. dense layer, softmax etc), whereas ensemble of a rule-based and SVM model counts only as '3 components'. Thus, we report only a trend of increasing architectural diversity, leaving a detailed analysis of the variability of the components as future work.

For example, novel systems based on BiLSTM are likely to be collocated with at least several other components, whereas ensemble of a rule-based and SVM model counts only as '3 components', while being possibly as complex as the former.

5.3 Relative frequency

We listed the top 10 most frequently mentioned architectural components in 2013 and 2019. This is depicted in Fig.2. As four of the components ('SVM', 'lower case', 'n-grams' and 'tokenization') were in both lists, we analysed 16 components: 'POS tagging', 'lexical features', 'normalization', 'lemmatization', 'BoW', 'disambiguation' (for 2013, dashdot lines), and 'Word Embeddings', 'LSTM', 'softmax', GloVe', 'CNN', 'embedding layer' (for 2019, dashed lines). These 4 mutual components (solid lines) show longevity, as their $freq_{rel}$ do not significantly change over the years, i.e. coefficients of linear regression fits stay in the $(-0.02, 0.02)$ range. Similarly, 'BoW' also maintains rather constant use ($coef = 0.0$). Highest drops are apparent for 'lemmatization' and 'disambiguation': -0.05 and -0.04 respectively. Although 'POS tagging' $coef = -0.05$, its $r^2 = 0.46$ suggests that the LR fit is not perfect, and in fact we observe a frequency increase along the 2012-2015 period, and then a drop until 2019. A significant popularity boost can be observed for 'LSTM', 'Word Embeddings', 'embedding layer', 'softmax' and 'GloVe'. They all have high positive $coef$ values and $r^2 > 0.9$. A list of top 100 components in 2019, with frequencies, $coef$, r^2 is presented in the Appendix.

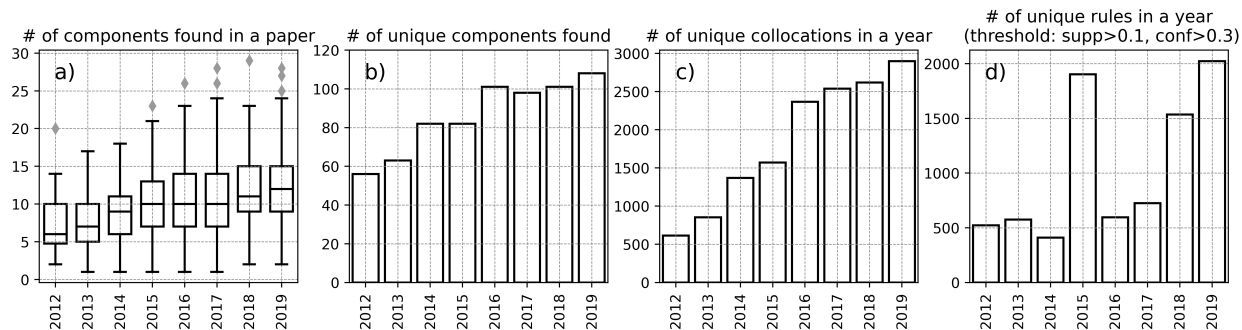


Figure 3: Statistics for SemEval 2012-2019: a) # of components found in a single paper; b) # of unique components; c) # of unique 2-component collocations d) # of unique rules found by Apriori algorithm.

5.4 Collocations

Similarly to the trend mentioned in 5.2, we observe a higher number of unique collocations found over the years (Fig.3c). This is expected as in general $\#components \sim \#collocations$. A relevant pattern is depicted in Fig.3d, where the # of unique rules in 2015 is significantly higher than in 2016-2018, not matching trends from former plots. Furthermore, despite higher # of collocations in 2014 than 2012-2013, again, the # of unique rules is lower.

Based on Fig.1 and Fig.3 a comparison between consecutive years can be performed. In 2014 more publications were submitted than in 2015 (134 vs. 118), but the # of components found is almost equal (1169 vs. 1206). This corresponds to the lower # of components found in a single paper on average (Fig.3a). At the same time, the # of unique components are equal (both 82, Fig.3b). The higher average # of components in a paper induces more possible collocations, and this is confirmed in Fig.3c (1370 vs 1570 2-element collocations). What draws the attention, is a prominent difference in the # of unique rules (409 vs 1902, Fig.3d). To explain it, we need to take a closer look into the Apriori algorithm and components distribution over the papers. Let N be the total # of components found, u the # of unique components, n the # of components in a paper and m the # of papers, assuming that $N/m < u$. Considering two extreme situations, having i) minimal and ii) maximal possible total # of collocations C . In i) all components are evenly distributed among papers. Thus, each paper contains $n = N/m$ components. The # of collocation in each paper equals $n!$, and $C = m \cdot n! = m \cdot (N/m)!$. In ii) we have x papers containing u components, and $(m - x)$ with only one component, where $x = (N - m)/(u - 1)$. Then $C = x \cdot u! = (N - m)/(u - 1) \cdot u!$. To sum up, in i) $C \sim (N/m)!$ and in ii) $C \sim u!$ and $N/m \ll u$ (e.g. for year 2014 $N/m = 1196/134 \cong 9 \ll u = 82$). Thus, the more uneven distribution of the components, the more collocations are expected. The simple intuition behind this is that one long list generates more collocations than several short lists. The same would apply to the rules generated by the Apriori algorithm, if not the thresholds $supp > 0.1$ and $conf > 0.3$. If in the set of papers there are only several papers with a high # of components, and a rule belongs to a large group of rules generated from such paper, it is likely that the rule does not pass the $supp$ threshold. Thus, the final # of unique rules compared with # of collocations reveals the components' distribution in the papers. High # in 2015 (Fig.3d) signifies higher variation of frequent collocations, comparing to 2014 (note, that # of unique components in 2014 and 2015 are \cong). In summary, there are many component-triples (or quadruples, up to octuples) consistently occurring in the papers (i.e. at least in 10% of papers). Similarly, in SemEval 2016-2018 less rules pass the threshold than in 2015, despite the higher # of components. It again signifies a higher variation of rules in 2015. Note, this reflects a general comparative analysis of components and collocations frequency, rather than a significance evaluation.

An analysis of two-element collocations is summarized in Fig.4, where the top 10 collocations for years 2012-2019 ranked by G^2 are depicted, together with corresponding frequencies and odds ratios. A high significance of some collocations is expected in advance. Firstly, for hyponym-hypernym pairs, e.g. IDF & term frequency (which is the top collocation 6 times), clustering & k-means, Brown & clustering or BiLSTM & LSTM the expectation of high significance is confirmed. Secondly, nearly inseparable pairs due to the model architecture, like CNN & pooling, pooling & softmax or decoder & encoder are also present in the summary. Finally, less expected but still significant collocations show variety and distinguishable trends over the years. From 2016, most of the top collocations consist of deep learning related components, however some of the older methods (like SVM & term frequency, skipgram & w2v) seem to maintain their longevity and relevance. Before 2016, a higher variation in the top10 lists is observed, as there is no obvious choice for best architecture among participating teams. Some of more notable collocations are: CRF & NER (Conditional random field & Named-entity recognition, in 2013, 2014, 2017), LSI (Latent semantic indexing) collocations: & PCA, & BoW, & semantic features, among others.

Architectures of Meaning

A Systematic Corpus Analysis of NLP Systems

Rank	2012				2013				2014				2015			
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α
1	IDF & term frequency	4	5.7	4.7	IDF & term frequency	5	15.7	13.2	IDF & term frequency	10	17.9	5.9	Autoencoder & RecNN	3	14.7	33.1
2	NLTK & stemming	4	4.6	3.9	disambiguation & word sense disambiguation	5	7.0	4.6	LSI & PCA	8	12.9	5.2	IDF & term frequency	7	12.7	5.9
3	LSI & PCA	5	4.3	3.1	CRF & NER	3	4.3	4.7	clustering & word clusters	13	11.3	3.1	Brown & clustering	12	11.6	3.4
4	PCA & Word Embeddings	3	4.0	4.4	dependency parsing & semantic features	14	4.0	1.9	Brown & word clusters	9	10.6	3.9	Bayes classifier & scikit-learn	4	11.2	10.5
5	dependency parsing & heuristic	6	3.7	2.6	NER & Stanford Parser	3	3.9	4.3	disambiguation & word sense disambiguation	4	10.0	9.2	clustering & word clusters	9	10.6	3.9
6	PCA & syntactic features	4	3.6	3.3	disambiguation & lexical features	3	3.8	4.2	Deep Learning & skipgram	3	9.8	13.9	dynamic pooling & pooling	3	10.3	14.9
7	BoW & LSI	4	3.5	3.2	Bayes classifier & feature selection	5	3.8	2.9	LSI & semantic features	11	9.6	3.1	alignment & semantic features	25	9.6	2.1
8	PMI & normalization	3	2.1	2.8	decision tree & semantic features	4	3.8	3.3	CRF & NER	12	9.6	3.0	SRL & semantic features	9	8.3	3.3
9	POS tagging & lexical features	5	2.0	2.1	lower case & stemming	6	3.5	2.5	Brown & clustering	13	9.1	2.7	NLTK & stemming	7	8.1	3.8
10	BoW & disambiguation	3	1.9	2.6	heuristic & semantic features	5	3.5	2.7	IG & feature selection	3	6.9	7.9	pooling & softmax	3	8.0	9.6
2016																
1	CNN & pooling	23	29.4	4.2	IDF & term frequency	21	26.6	4.2	IDF & term frequency	22	33.6	5.0	IDF & term frequency	47	46.1	3.4
2	IDF & term frequency	23	26.8	3.9	CNN & pooling	28	22.6	3.0	FNN & fully connected layers	13	13.8	3.6	BERT & transformer	27	43.4	5.2
3	pooling & softmax	20	20.4	3.5	LSTM & RNN	23	19.6	3.1	CNN & pooling	28	13.5	2.3	CNN & pooling	40	18.8	2.2
4	LSTM & RNN	14	20.0	4.6	CNN & softmax	25	15.3	2.5	DL & fully connected layers	6	13.0	7.3	SVM & term frequency	41	18.2	2.2
5	CNN & softmax	20	18.0	3.2	fully connected layer & pooling	18	15.3	3.1	Brown & clustering	7	12.1	5.5	clustering & k-means	4	16.9	22.5
6	GRU & RNN	9	15.6	5.6	BILSTM & LSTM	13	14.9	3.8	GRU & RNN	13	11.5	3.1	BILSTM & LSTM	58	16.7	1.8
7	Brown & clustering	17	15.2	3.2	pooling & softmax	23	13.9	2.5	SVM & scikit-learn	21	11.3	2.4	BERT & encoder	22	16.6	2.8
8	fully connected layer & softmax	13	12.5	3.3	CNN & fully connected layer	18	13.4	2.8	decoder & encoder	5	11.2	7.6	encoder & transformer	19	16.6	3.1
9	BILSTM & LSTM	5	11.6	8.0	fully connected layer & softmax	17	12.3	2.8	stemming & stopword removal	8	11.2	4.5	IDF & SVM	38	16.3	2.1
10	Deep Learning & fully connected layer	9	11.2	4.0	CRF & NER	8	12.1	4.8	skipgram & w2v	24	11.1	2.2	SVM & scikit-learn	29	15.5	2.4

Figure 4: Top 10 two-component collocations in SemEval (SA, SEM and IE tasks) for years 2012-2019 ranked by G^2 (only positive association: $odds\ ratio\alpha > 1$).

Investigating collocations of 3 or more, we summarized top 10 sets of components for each year, derived from association rules in Fig.5. In the last 3 years most significant collocations consist of 'LSTM' and/or 'Word Embeddings', with also high visibility of 'CNN' and/or 'tokenization'. In years 2012-2015 various collocations of 'POS tagging' and 'tokenization' are noticeable, which interestingly corresponds to $freq_{rel}$ but not to collocations from Fig.4. Apart from the mentioned above, some more of the recurring collocations in this summary are: deep learning components ('softmax', 'pooling'), 'n-gram', 'lemmatization' and 'IDF' collocations.

5.5 Network graphs

Summaries discussed above (of 2 and ≥ 3 components collocations) are limited to 10 rows. In order to deliver more specific and concise insights about frequencies and collocations, we visualised the top 100 collocations and presented as network graphs in Fig.6. Note, that most of the observations mentioned in previous paragraphs are confirmed by graphs. Additionally, node size \sim frequency, red color intensity \sim node degree (# of edges connected to the node) and edge width $\sim G^2$ simultaneously inform about the component popularity, the density and significance of connections. Each graph consists of 100 edges, but various # of nodes.

The network graph for SemEval 2012 is relatively dense, which reflects the fact that in this year only 56 unique components and 613 unique collocations were found. It consist of only 28 nodes, and the high frequency of a component corresponds to the high degree of the node, i.e. 'Semantic features', 'POS tagging', 'tokenization'. Significant edges (collocations) can be found both between high/low frequency/degree nodes. Note, that in 2012 only the SEM task type is reported. In 2013, the graph consists of 41 nodes and central roles play 'POS tagging', 'n-grams', 'lemmatization' and 'tokenization'. 'SVM', 'normalization' and 'lexical features' also appear to be highly relevant representational components. In 2014, again there is an increase of variety (51 nodes). Still, 'POS tagging', 'lexical features', 'n-grams', 'tokenization' are the most noticeable, followed by 'SVM', 'lemmatization' and 'stopword removal'. We observe a growing number of less frequent components.

In the SemEval 2015 graph (53 nodes), we observe a noticeable split between architectural patterns. In the first cluster, we have again 'lexical' and 'semantic features', 'POS tagging', 'lemmatization', 'n-grams' and 'SVM'. In the other,

Architectures of Meaning

A Systematic Corpus Analysis of NLP Systems

Rank	2012				2013				2014				2015																																																																																																																																																																																																														
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α																																																																																																																																																																																																											
1	POS tagging, lemmatization, tokenization	0.205	57	18.0	POS tagging, normalization, tokenization	0.261	95	4.5	POS tagging, SVM, n-grams	0.239	97	6.8	POS tagging, SVM, tokenization	0.339	228	2.7																																																																																																																																																																																																											
2	normalization, stopword removal, tokenization	0.159	44	11.5	POS tagging, lemmatization, tokenization	0.25	93	5.7	POS tagging, lexical features, n-grams	0.261	96	5.2	lexical features, n-grams, tokenization	0.297	218	5.9																																																																																																																																																																																																											
3	alignment, normalization, tokenization	0.159	44	11.5	POS tagging, SVM, tokenization	0.216	76	4.0	POS tagging, SVM, lexical features, n-grams	0.194	94	12.7	POS tagging, n-grams, tokenization	0.339	218	1.8																																																																																																																																																																																																											
4	POS tagging, alignment, lemmatization, tokenization	0.136	43	23.2	POS tagging, n-grams, tokenization	0.239	76	2.3	POS tagging, SVM, lexical features	0.231	84	6.6	POS tagging, SVM, n-grams	0.314	213	3.2																																																																																																																																																																																																											
5	POS tagging, lemmatization, normalization, stopword removal, tokenization	0.114	43	84.4	lexical features, n-grams, tokenization	0.205	73	4.0	SVM, lexical features, n-grams	0.209	81	6.5	SVM, n-grams, tokenization	0.305	209	3.5																																																																																																																																																																																																											
6	POS tagging, alignment, lemmatization	0.159	42	11.8	POS tagging, lemmatization, normalization	0.205	72	4.2	n-grams, normalization, tokenization	0.187	73	5.6	POS tagging, lexical features, n-grams	0.288	203	4.3																																																																																																																																																																																																											
7	alignment, lemmatization, tokenization	0.159	42	10.5	POS tagging, lexical features, tokenization	0.216	70	2.5	lexical features, n-grams, tokenization	0.216	70	3.3	SVM, lexical features, n-grams	0.263	198	9.3																																																																																																																																																																																																											
8	alignment, semantic features, tokenization	0.159	41	7.4	n-grams, normalization, tokenization	0.193	69	4.3	lexical features, n-grams, normalization	0.149	69	23.0	POS tagging, lexical features, tokenization	0.288	187	2.2																																																																																																																																																																																																											
9	lemmatization, semantic features, stopword removal	0.159	41	7.4	BoW, POS tagging, tokenization	0.159	65	13.2	POS tagging, lexical features, tokenization	0.246	67	2.0	POS tagging, SVM, n-grams, tokenization	0.263	186	4.6																																																																																																																																																																																																											
10	lemmatization, normalization, semantic features	0.159	40	6.4	POS tagging, lemmatization, normalization, tokenization	0.159	61	9.0	lexical features, n-grams, normalization, tokenization	0.127	66	15.2	POS tagging, lemmatization, tokenization	0.254	174	3.8																																																																																																																																																																																																											
<table border="1"> <thead> <tr> <th rowspan="2">Rank</th> <th colspan="4">2016</th> <th colspan="4">2017</th> <th colspan="4">2018</th> <th colspan="4">2019</th> </tr> <tr> <th>Collocation</th> <th>supp</th> <th>G²</th> <th>α</th> <th>Collocation</th> <th>supp</th> <th>G²</th> <th>α</th> <th>Collocation</th> <th>supp</th> <th>G²</th> <th>α</th> <th>Collocation</th> <th>supp</th> <th>G²</th> <th>α</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>CNN, Word Embeddings, pooling</td> <td>0.131</td> <td>142</td> <td>193.9</td> <td>Word Embeddings, tokenization, w2v</td> <td>0.255</td> <td>151</td> <td>10.2</td> <td>BiLSTM, LSTM, Word Embeddings</td> <td>0.229</td> <td>225</td> <td>25.0</td> <td>BiLSTM, LSTM, Word Embeddings</td> <td>0.268</td> <td>347.7</td> <td>36.4</td> </tr> <tr> <td>2</td> <td>CNN, pooling, softmax</td> <td>0.113</td> <td>128</td> <td>278.8</td> <td>CNN, Word Embeddings, pooling</td> <td>0.172</td> <td>149</td> <td>37.0</td> <td>LSTM, Word Embeddings, tokenization</td> <td>0.274</td> <td>208</td> <td>4.2</td> <td>LSTM, Word Embeddings, softmax</td> <td>0.305</td> <td>331.3</td> <td>9.7</td> </tr> <tr> <td>3</td> <td>CNN, Word Embeddings, pooling, softmax</td> <td>0.107</td> <td>121</td> <td>191.1</td> <td>LSTM, RNN, Word Embeddings</td> <td>0.14</td> <td>141</td> <td>93.6</td> <td>Word Embeddings, tokenization, w2v</td> <td>0.255</td> <td>201</td> <td>7.8</td> <td>LSTM, Word Embeddings, tokenization</td> <td>0.291</td> <td>320.1</td> <td>13.3</td> </tr> <tr> <td>4</td> <td>POS tagging, lexical features, tokenization</td> <td>0.214</td> <td>116</td> <td>8.6</td> <td>CNN, pooling, softmax</td> <td>0.134</td> <td>121</td> <td>58.4</td> <td>LSTM, Word Embeddings, softmax</td> <td>0.236</td> <td>179</td> <td>3.7</td> <td>LSTM, Word Embeddings, embedding layer</td> <td>0.268</td> <td>304.9</td> <td>8.0</td> </tr> <tr> <td>5</td> <td>IDF, n-grams, term frequency</td> <td>0.107</td> <td>115</td> <td>111.0</td> <td>CNN, Word Embeddings, softmax</td> <td>0.153</td> <td>115</td> <td>14.9</td> <td>LSTM, RNN, Word Embeddings</td> <td>0.210</td> <td>179</td> <td>7.6</td> <td>GloVe, LSTM, Word Embeddings</td> <td>0.277</td> <td>285.1</td> <td>3.9</td> </tr> <tr> <td>6</td> <td>Word Embeddings, tokenization, w2v</td> <td>0.208</td> <td>109</td> <td>5.1</td> <td>Word Embeddings, n-grams, w2v</td> <td>0.204</td> <td>115</td> <td>7.3</td> <td>Word Embeddings, softmax, tokenization</td> <td>0.229</td> <td>179</td> <td>6.7</td> <td>IDF, SVM, term frequency</td> <td>0.164</td> <td>258.4</td> <td>79.1</td> </tr> <tr> <td>7</td> <td>POS tagging, lemmatization, tokenization</td> <td>0.208</td> <td>108</td> <td>6.5</td> <td>CNN, Word Embeddings, pooling, softmax</td> <td>0.127</td> <td>114</td> <td>52.9</td> <td>LSTM, Word Embeddings, embedding layer</td> <td>0.197</td> <td>170</td> <td>7.9</td> <td>IDF, Word Embeddings, term frequency</td> <td>0.160</td> <td>257.1</td> <td>79.1</td> </tr> <tr> <td>8</td> <td>Word Embeddings, pooling, softmax</td> <td>0.113</td> <td>101</td> <td>36.6</td> <td>GloVe, Word Embeddings, tokenization</td> <td>0.185</td> <td>114</td> <td>7.6</td> <td>LSTM, Word Embeddings, attention layer</td> <td>0.185</td> <td>169</td> <td>13.0</td> <td>Word Embeddings, softmax, tokenization</td> <td>0.225</td> <td>246.8</td> <td>16.8</td> </tr> <tr> <td>9</td> <td>POS tagging, lexical features, n-grams</td> <td>0.196</td> <td>100</td> <td>6.0</td> <td>CNN, Word Embeddings, pooling, tokenization</td> <td>0.134</td> <td>113</td> <td>24.6</td> <td>CNN, Word Embeddings, pooling</td> <td>0.172</td> <td>165</td> <td>14.7</td> <td>BiLSTM, LSTM, softmax</td> <td>0.192</td> <td>244.4</td> <td>47.5</td> </tr> <tr> <td>10</td> <td>POS tagging, n-grams, tokenization</td> <td>0.232</td> <td>96</td> <td>2.4</td> <td>IDF, Word Embeddings, term frequency</td> <td>0.108</td> <td>109</td> <td>64.2</td> <td>CNN, Word Embeddings, tokenization</td> <td>0.197</td> <td>157</td> <td>10.4</td> <td>GloVe, Word Embeddings, softmax</td> <td>0.221</td> <td>241.1</td> <td>16.3</td> </tr> </tbody> </table>																	Rank	2016				2017				2018				2019				Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	1	CNN, Word Embeddings, pooling	0.131	142	193.9	Word Embeddings, tokenization, w2v	0.255	151	10.2	BiLSTM, LSTM, Word Embeddings	0.229	225	25.0	BiLSTM, LSTM, Word Embeddings	0.268	347.7	36.4	2	CNN, pooling, softmax	0.113	128	278.8	CNN, Word Embeddings, pooling	0.172	149	37.0	LSTM, Word Embeddings, tokenization	0.274	208	4.2	LSTM, Word Embeddings, softmax	0.305	331.3	9.7	3	CNN, Word Embeddings, pooling, softmax	0.107	121	191.1	LSTM, RNN, Word Embeddings	0.14	141	93.6	Word Embeddings, tokenization, w2v	0.255	201	7.8	LSTM, Word Embeddings, tokenization	0.291	320.1	13.3	4	POS tagging, lexical features, tokenization	0.214	116	8.6	CNN, pooling, softmax	0.134	121	58.4	LSTM, Word Embeddings, softmax	0.236	179	3.7	LSTM, Word Embeddings, embedding layer	0.268	304.9	8.0	5	IDF, n-grams, term frequency	0.107	115	111.0	CNN, Word Embeddings, softmax	0.153	115	14.9	LSTM, RNN, Word Embeddings	0.210	179	7.6	GloVe, LSTM, Word Embeddings	0.277	285.1	3.9	6	Word Embeddings, tokenization, w2v	0.208	109	5.1	Word Embeddings, n-grams, w2v	0.204	115	7.3	Word Embeddings, softmax, tokenization	0.229	179	6.7	IDF, SVM, term frequency	0.164	258.4	79.1	7	POS tagging, lemmatization, tokenization	0.208	108	6.5	CNN, Word Embeddings, pooling, softmax	0.127	114	52.9	LSTM, Word Embeddings, embedding layer	0.197	170	7.9	IDF, Word Embeddings, term frequency	0.160	257.1	79.1	8	Word Embeddings, pooling, softmax	0.113	101	36.6	GloVe, Word Embeddings, tokenization	0.185	114	7.6	LSTM, Word Embeddings, attention layer	0.185	169	13.0	Word Embeddings, softmax, tokenization	0.225	246.8	16.8	9	POS tagging, lexical features, n-grams	0.196	100	6.0	CNN, Word Embeddings, pooling, tokenization	0.134	113	24.6	CNN, Word Embeddings, pooling	0.172	165	14.7	BiLSTM, LSTM, softmax	0.192	244.4	47.5	10	POS tagging, n-grams, tokenization	0.232	96	2.4	IDF, Word Embeddings, term frequency	0.108	109	64.2	CNN, Word Embeddings, tokenization	0.197	157	10.4	GloVe, Word Embeddings, softmax	0.221	241.1	16.3
Rank	2016				2017				2018				2019																																																																																																																																																																																																														
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α																																																																																																																																																																																																											
1	CNN, Word Embeddings, pooling	0.131	142	193.9	Word Embeddings, tokenization, w2v	0.255	151	10.2	BiLSTM, LSTM, Word Embeddings	0.229	225	25.0	BiLSTM, LSTM, Word Embeddings	0.268	347.7	36.4																																																																																																																																																																																																											
2	CNN, pooling, softmax	0.113	128	278.8	CNN, Word Embeddings, pooling	0.172	149	37.0	LSTM, Word Embeddings, tokenization	0.274	208	4.2	LSTM, Word Embeddings, softmax	0.305	331.3	9.7																																																																																																																																																																																																											
3	CNN, Word Embeddings, pooling, softmax	0.107	121	191.1	LSTM, RNN, Word Embeddings	0.14	141	93.6	Word Embeddings, tokenization, w2v	0.255	201	7.8	LSTM, Word Embeddings, tokenization	0.291	320.1	13.3																																																																																																																																																																																																											
4	POS tagging, lexical features, tokenization	0.214	116	8.6	CNN, pooling, softmax	0.134	121	58.4	LSTM, Word Embeddings, softmax	0.236	179	3.7	LSTM, Word Embeddings, embedding layer	0.268	304.9	8.0																																																																																																																																																																																																											
5	IDF, n-grams, term frequency	0.107	115	111.0	CNN, Word Embeddings, softmax	0.153	115	14.9	LSTM, RNN, Word Embeddings	0.210	179	7.6	GloVe, LSTM, Word Embeddings	0.277	285.1	3.9																																																																																																																																																																																																											
6	Word Embeddings, tokenization, w2v	0.208	109	5.1	Word Embeddings, n-grams, w2v	0.204	115	7.3	Word Embeddings, softmax, tokenization	0.229	179	6.7	IDF, SVM, term frequency	0.164	258.4	79.1																																																																																																																																																																																																											
7	POS tagging, lemmatization, tokenization	0.208	108	6.5	CNN, Word Embeddings, pooling, softmax	0.127	114	52.9	LSTM, Word Embeddings, embedding layer	0.197	170	7.9	IDF, Word Embeddings, term frequency	0.160	257.1	79.1																																																																																																																																																																																																											
8	Word Embeddings, pooling, softmax	0.113	101	36.6	GloVe, Word Embeddings, tokenization	0.185	114	7.6	LSTM, Word Embeddings, attention layer	0.185	169	13.0	Word Embeddings, softmax, tokenization	0.225	246.8	16.8																																																																																																																																																																																																											
9	POS tagging, lexical features, n-grams	0.196	100	6.0	CNN, Word Embeddings, pooling, tokenization	0.134	113	24.6	CNN, Word Embeddings, pooling	0.172	165	14.7	BiLSTM, LSTM, softmax	0.192	244.4	47.5																																																																																																																																																																																																											
10	POS tagging, n-grams, tokenization	0.232	96	2.4	IDF, Word Embeddings, term frequency	0.108	109	64.2	CNN, Word Embeddings, tokenization	0.197	157	10.4	GloVe, Word Embeddings, softmax	0.221	241.1	16.3																																																																																																																																																																																																											

Figure 5: Top 10 collocations in SemEval (SA, SEM and IE tasks) derived from rules found using Apriori for years 2012-2019, ranked by G^2 (only positive association: *odds ratio* $\alpha > 1$).

'Word Embeddings' with connections to 'w2v', 'CNN', 'softmax' etc. are present, this represents a transition point in the architectural patterns in the direction of neural representation.

In 2016, we observe a substantial increase in # of unique components and collocations found, as well as in total # of components, likewise the variability in the graph: # of nodes increased to 67. 'Word Embedding' as a center of neural network related nodes, draws the attention to high interest of deep learning and emergence of LSTM usage. However, canonical NLP components (features and pre-processing steps) are still highly present: 'POS tagging', 'lexical features', 'n-grams' or 'lemmatization' as well as 'SVM'.

For 2017, the center of the graph is populated by deep learning components, i.e. 'LSTM', 'CNN', 'softmax'. However, ≥ 3 collocations from Fig.5 are represented by the upper left branch of the graph, with 'Word Embedding', 'tokenization' and 'w2v'. In 2018 we observe the prevalence of sequence-based DL models ('LSTM', 'RNN' at the center). 'Tokenization' and 'POS tagging' are still frequent. A noticeable change is that we observe 'SVM' with other ML methods like 'RF', 'Bayes classifier', 'decision tree' to constitute the upper part of the graph. In the lower part the connection between 'w2v' and 'skipgram' can be highlighted.

Finally, the SemEval 2019, which is dominated by deep learning, is represented by a graph which is denser than the previous 3 editions (56 nodes, 11 and 8 less than in 2017 and 2018). Apart from the most frequent 'Word Embeddings', which is interestingly connected only to 'LSTM', there are still some old ML components like 'SVM', 'logistic regression' or 'Bayes classifier'. Upper right branch consists of 'POS tagging', 'lexical' and 'semantic features'. Upper left part highlights the usage of 'BERT' and 'ELMO'. Similarly to 2018, 'attention layer' and 'n-grams' are close to the center. Note, that for the 2019 graph we mainly report components associated with the SA task.

In summary, graphs illustrate broader collocation patterns (≥ 3) (Fig.5). They provide a natural normalisation mechanism for putting high-frequency canonical components into context (although term frequency and IDF appear to be very significant in 2-component collocations, they weaken in ≥ 3 collocations, and play rather marginal role

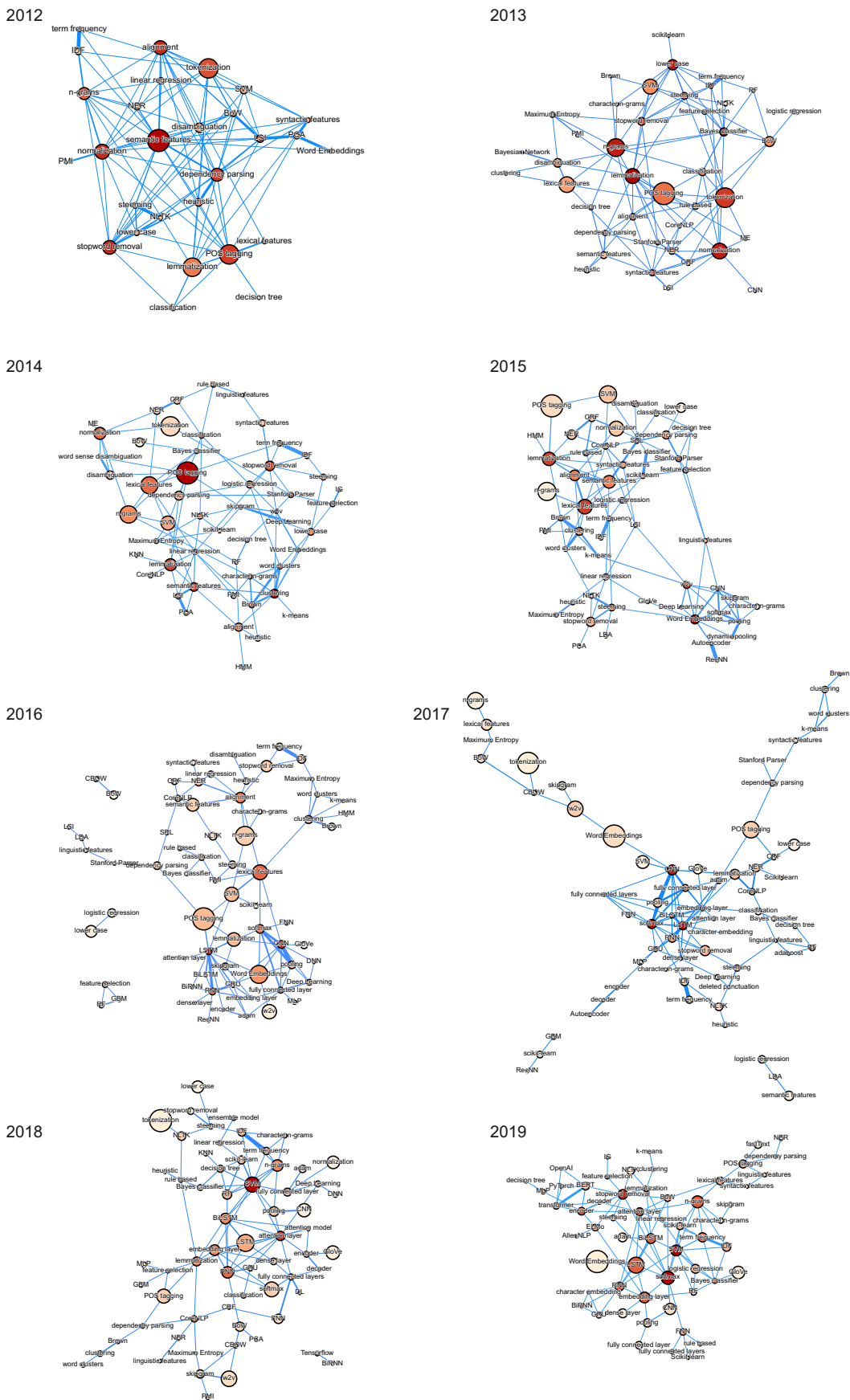


Figure 6: Network graphs built using the top 100 collocations from SA, SEM and IE tasks in SemEval from 2012 to 2019. Node size is proportional to the frequency of the component, the red color intensity to node degree, edge width to G^2 of the collocation. For full resolution see the digital version of the paper.

when analysing the graphs). Moreover they provide a natural way to observe the emergence of clusters of architectural components.

Network collocation graphs built over collocation analysis can support the interpretation of architectural patterns at scale. The analysis confirmed architectural patterns coherent with the major trends on the evolution of the field. The proposed method excelled at inducing functional clusters. Among others, one observable pattern is on the longevity and the cumulative aspect of some architectural motifs, which tend to consolidate and compose with novel functional clusters (e.g. the emerging and recurring architectural theme: POS, lemma – > embeddings – > Sequence DL architectures – > attention – > transformers).

The reader is referred to a task-based detailed results for SA, SEM and IE see Supplementary Material.

6 Related work

To the best of our knowledge this is the first time that corpus analysis methods were developed to perform a systematic analysis of architectural patterns in NLP and AI or applied for a systematic survey in NLP. In the space of corpus linguistics, Gries [6],[7] provides a comprehensive reference for statistical corpus analysis, which we build upon, extending it with a saturation-based lexicon construction methods and with association rule and graph collocation mining. Other comprehensive studies presenting a perspective on corpus linguistics are provided by Hall [8], Brezina [5], Mohammad [12][11] and Bollmann [3]. On the systematic literature analysis, Wysocki [15] describes a comprehensive critical structured survey of the Semeval campaign, focusing on the distribution of task types and their impact. Methodologically, these works emphasise corpus statistics to some extent but the analyses do not provide a granular architectural perspective.

Surveys targeted on specific task types such as Question Answering [4] or Sentiment Analysis [10] or a specific architectural paradigm [16] are abundant but most commonly they do not apply any statistical analysis method and focus on smaller scale text interpretation (in contrast to a corpus analysis method which can scale up to large reference bases).

7 Conclusions

This paper proposed a novel statistical corpus analysis framework targeted towards the interpretation of NLP architectural patterns at scale. The framework combines a saturation-based lexicon construction, statistical collocation methods and graph collocations to derive an aggregate representation of architectural patterns. The framework was validated in the context of the corpus analysis of Semeval tasks and demonstrated consistent architectural patterns which can be used to address questions on the evolution of NLP architectures in a data-driven manner.

References

- [1] Rakesh Agrawal and Ramakrishnan Srikant. “Fast Algorithms for Mining Association Rules in Large Databases”. In: *Proceedings of the 20th International Conference on Very Large Data Bases*. VLDB ’94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, pp. 487–499. ISBN: 1558601538.
- [2] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. *Gephi: An Open Source Software for Exploring and Manipulating Networks*. 2009. URL: <http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154>.
- [3] Marcel Bollmann and Desmond Elliott. “On Forgetting to Cite Older Papers: An Analysis of the ACL Anthology”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 7819–7827. DOI: 10.18653/v1/2020.acl-main.699. URL: <https://www.aclweb.org/anthology/2020.acl-main.699>.
- [4] Abdelghani Bouziane et al. “Question answering systems: survey and trends”. In: *Procedia Computer Science* 73 (2015), pp. 366–375.
- [5] Vaclav Brezina. *Statistics in Corpus Linguistics: A Practical Guide*. Cambridge University Press, 2018. DOI: 10.1017/9781316410899.
- [6] St Gries. “Useful statistics for corpus linguistics”. In: *A mosaic of corpus linguistics: selected approaches* (2010), pp. 269–291. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.160.9846%7B%5C%7Drep=rep1%7B%5C%7Dtype=pdf>.

- [7] Stefan Th. Gries. “Quantitative designs and statistical techniques”. In: *The Cambridge Handbook of English Corpus Linguistics*. Ed. by Douglas Biber and Randi Editors Reppen. Cambridge Handbooks in Language and Linguistics. Cambridge University Press, 2015, pp. 50–72. DOI: 10.1017/CBO9781139764377.004.
- [8] David Hall, Daniel Jurafsky, and Christopher D. Manning. “Studying the History of Ideas Using Topic Models”. In: *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Hawaii: Association for Computational Linguistics, Oct. 2008, pp. 363–371. URL: <https://www.aclweb.org/anthology/D08-1038>.
- [9] Mathieu Jacomy et al. “ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software”. In: *PLOS ONE* 9.6 (June 2014), pp. 1–12. DOI: 10.1371/journal.pone.0098679. URL: <https://doi.org/10.1371/journal.pone.0098679>.
- [10] Walaa Medhat, Ahmed Hassan, and Hoda Korashy. “Sentiment analysis algorithms and applications: A survey”. In: *Ain Shams engineering journal* 5.4 (2014), pp. 1093–1113.
- [11] Saif M. Mohammad. “NLP Scholar: A Dataset for Examining the State of NLP Research”. English. In: *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 868–877. ISBN: 979-10-95546-34-4. URL: <https://www.aclweb.org/anthology/2020.lrec-1.109>.
- [12] Saif M. Mohammad. *The State of NLP Literature: A Diachronic Analysis of the ACL Anthology*. 2019. arXiv: 1911.03562 [cs.DL].
- [13] Tommy Odland. *Efficient-Apriori*. 2018. URL: <https://github.com/tommyod/Efficient-Apriori#efficient-apriori----->.
- [14] Pang-Ning Tan, Vipin Kumar, and Jaideep Srivastava. “Selecting the right objective measure for association analysis”. In: *Information Systems* 29.4 (2004). Knowledge Discovery and Data Mining (KDD 2002), pp. 293–313. ISSN: 0306-4379. DOI: [https://doi.org/10.1016/S0306-4379\(03\)00072-3](https://doi.org/10.1016/S0306-4379(03)00072-3). URL: <http://www.sciencedirect.com/science/article/pii/S0306437903000723>.
- [15] Oskar Wysocki, Malina Florea, and Andre Freitas. *What is SemEval evaluating? A Systematic Analysis of Evaluation Campaigns in NLP*. 2020. arXiv: 2005.14299 [cs.CL].
- [16] Lei Zhang, Shuai Wang, and Bing Liu. “Deep learning for sentiment analysis: A survey”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4 (2018), e1253.

8 Supplementary Material

Rank	2013				2014				2015							
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α				
1	Bayes classifier & classification	6	4.0	2.7	Brown & word clusters	5	7.1	4.7	NLTK & steeming	4	6.5	5.3				
2	Bayes classifier & NLTK	4	2.6	2.6	CRF & NER	5	6.1	4.1	clustering & word clusters	8	4.4	2.4				
3	Bayes classifier & normalization	4	2.4	0.5	character n-grams & word clusters	5	6.0	4.0	skipgram & w2v	4	4.4	3.7				
4	SVM & feature selection	6	2.1	2.0	NER & semantic features	4	6.0	4.8	softmax & w2v	4	4.3	3.6				
5	lexical features & rule based	5	1.8	2.0	character n-grams & clustering	7	5.9	3.1	Brown & clustering	8	3.4	2.2				
6	lower case & tokenization	14	1.5	1.4	clustering & word clusters	8	5.8	2.8	CNN & Word Embeddings	5	3.3	2.7				
7	POS tagging & feature selection	6	1.2	1.7	Brown & character n-grams	4	5.6	4.6	CNN & w2v	4	3.2	3.0				
8	lemmatization & normalization	9	1.1	1.5	stopword removal & term frequency	4	5.3	4.3	Word Embeddings & w2v	7	3.1	2.2				
9	Bayes classifier & feature selection	4	1.1	1.8	Brown & clustering	6	4.9	3.0	steeming & word clusters	4	2.6	2.6				
10	disambiguation & lemmatization	5	1.1	1.7	Stanford Parser & dependency parsing	7	4.5	2.6	Word Embeddings & softmax	4	2.5	2.5				
	2016				2017				2018				2019			
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α
1	IDF & term frequency	12	17.3	4.7	LSTM & RNN	14	15.0	3.7	IDF & term frequency	17	25.2	4.8	IDF & term frequency	39	36.8	3.3
2	CNN & pooling	17	15.2	3.2	IDF & term frequency	10	12.5	4.1	CNN & pooling	17	10.7	2.6	BERT & transformer	15	30.5	6.8
3	GRU & RNN	6	13.4	7.7	fully connected layer & softmax	12	10.8	3.2	CoreNLP & Maximum Entropy	4	10.4	9.4	CNN & pooling	32	16.9	2.4
4	CNN & softmax	15	12.9	3.1	CNN & pooling	16	10.3	2.6	skipgram & w2v	15	10.3	2.7	SVM & term frequency	35	13.9	2.1
5	pooling & softmax	14	12.7	3.2	CNN & softmax	14	9.5	2.7	RF & scikit-learn	13	9.5	2.8	SVM & scikit-learn	29	13.5	2.2
6	LSTM & RNN	6	11.2	6.1	CNN & fully connected layer	12	8.0	2.7	GRU & RNN	11	9.3	3.1	encoder & transformer	11	12.5	3.8
7	RNN & embedding layer	4	8.8	7.4	pooling & softmax	13	7.8	2.5	NLTK & rule based	4	7.4	6.1	IDF & SVM	33	12.4	2.0
8	GRU & LSTM	4	8.5	7.1	NLTK & heuristic	4	6.6	5.4	CoreNLP & NER	5	7.4	4.7	PyTorch & transformer	6	11.2	6.0
9	Deep Learning & MLP	4	7.5	6.2	BiLSTM & LSTM	5	6.6	4.3	steeming & stopword removal	7	7.3	3.5	BiLSTM & LSTM	44	10.2	1.7
10	CNN & Word Embeddings	20	7.2	2.0	heuristic & lemmatization	4	6.5	5.2	RF & decision tree	9	7.1	2.9	BERT & encoder	12	9.4	2.9

Supplementary Figure 1: Sentiment Analysis tasks in SemEval 2013-2019: Top 10 two-component collocations ranked by G^2 (only positive association: odds ratio $\alpha > 1$)

Rank	2013				2014				2015							
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α				
1	lexical features, n-grams, tokenization	0.39	132	5.2	POS tagging, lexical features, n-grams	0.5	232	5.9	lexical features, n-grams, tokenization	0.591	287	13.5				
2	POS tagging, lexical features, tokenization	0.341	110	2.8	POS tagging, SVM, lexical features	0.448	210	10.0	POS tagging, lexical features, n-grams	0.545	255	3.9				
3	lexical features, n-grams, normalization	0.317	107	6.2	SVM, lexical features, n-grams	0.414	192	5.8	POS tagging, lexical features, tokenization	0.545	249	2.1				
4	lexical features, normalization, tokenization	0.317	107	4.6	POS tagging, lexical features, tokenization	0.414	192	8.5	SVM, lexical features, tokenization	0.5	238	7.1				
5	lower case, n-grams, tokenization	0.293	106	10.6	POS tagging, SVM, n-grams	0.397	186	6.9	SVM, lexical features, n-grams	0.477	227	6.9				
6	SVM, n-grams, tokenization	0.293	106	15.3	POS tagging, SVM, lexical features, n-grams	0.379	179	6.9	SVM, lexical features, n-grams, tokenization	0.455	224	20.8				
7	POS tagging, lexical features, normalization	0.317	104	3.7	lexical features, n-grams, tokenization	0.345	157	6.1	SVM, n-grams, tokenization	0.455	219	9.7				
8	POS tagging, SVM, tokenization	0.293	101	7.3	SVM, lexical features, tokenization	0.31	139	5.1	POS tagging, lexical features, n-grams, tokenization	0.455	213	4.9				
9	n-grams, normalization, tokenization	0.293	101	7.3	POS tagging, lexical features, n-grams, tokenization	0.31	138	4.0	POS tagging, n-grams, tokenization	0.455	211	4.0				
10	POS tagging, lexical features, n-grams	0.317	99	1.7	POS tagging, n-grams, tokenization	0.31	132	1.9	POS tagging, SVM, lexical features	0.455	209	3.3				
	2016				2017				2018				2019			
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α
1	POS tagging, lexical features, tokenization	0.333	215	27.0	Word Embeddings, tokenization, w2v	0.299	174	9.2	LSTM, Word Embeddings, tokenization	0.33	260	5.7	LSTM, Word Embeddings, embedding layer	0.294	329.5	10.3
2	POS tagging, lexical features, n-grams	0.359	215	5.7	LSTM, RNN, Word Embeddings	0.194	153	300.6	Word Embeddings, lower case, tokenization	0.289	231	6.6	LSTM, Word Embeddings, softmax	0.301	318.2	11.6
3	POS tagging, n-grams, tokenization	0.359	208	3.8	CNN, Word Embeddings, pooling	0.224	144	16.8	BiLSTM, LSTM, Word Embeddings	0.268	246	55.0	BiLSTM, LSTM, Word Embeddings	0.264	317.7	37.2
4	POS tagging, lexical features, n-grams, tokenization	0.295	193	14.8	Word Embeddings, lexical features, w2v	0.239	140	7.6	LSTM, Word Embeddings, softmax	0.268	210	4.5	GloVe, LSTM, Word Embeddings	0.288	294.9	4.6
5	lexical features, n-grams, tokenization	0.308	185	6.9	Word Embeddings, n-grams, w2v	0.239	140	12.7	LSTM, Word Embeddings, embedding layer	0.237	204	14.5	LSTM, Word Embeddings, tokenization	0.301	290	2.8
6	CNN, Word Embeddings, pooling	0.205	165	163.7	Word Embeddings, pooling, tokenization	0.239	139	6.5	Word Embeddings, tokenization, w2v	0.247	198	10.9	IDF, SVM, term frequency	0.184	258.6	71.9
7	POS tagging, SVM, n-grams	0.282	164	4.3	GloVe, Word Embeddings, tokenization	0.209	133	26.9	LSTM, Word Embeddings, lower case	0.247	198	10.9	CNN, Word Embeddings, pooling	0.196	250.2	24.5
8	SVM, lexical features, n-grams	0.269	159	5.4	SVM, lexical features, n-grams	0.209	126	15.3	Word Embeddings, lexical features, tokenization	0.258	197	3.8	IDF, Word Embeddings, term frequency	0.172	249.3	108.8
9	POS tagging, SVM, lexical features	0.256	147	3.8	CNN, Word Embeddings, tokenization	0.224	123	3.8	LSTM, Word Embeddings, attention layer	0.227	194	13.3	BiLSTM, LSTM, softmax	0.196	233.6	41.1
10	POS tagging, clustering, lexical features	0.218	147	31.4	CNN, Word Embeddings, softmax	0.194	123	14.5	CNN, LSTM, Word Embeddings	0.227	181	9.6	IDF, n-grams, term frequency	0.166	233	79.7

Supplementary Figure 2: Sentiment Analysis tasks in SemEval 2013-2019: Top 10 collocations of ≥ 3 -components derived from rules found using Apriori algorithm, ranked by G^2 (only positive association: odds ratio $\alpha > 1$)

Rank	2012				2013				2014				2015			
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α
1	IDF & term frequency	4	5.4	4.5	POS tagging & lemmatization	10	1.9	1.7	POS tagging & dependency parsing	13	7.2	2.5	CRF & normalization	5	4.6	3.3
2	PCA & syntactic features	4	4.4	3.8	lemmatization & stopword removal	5	1.4	1.9	IDF & term frequency	6	5.6	3.3	lexical features & logistic regression	4	4.5	3.8
3	NLTK & steeming	4	4.4	3.7	POS tagging & tokenization	8	1.1	1.5	POS tagging & lemmatization	18	4.9	1.8	dependency parsing & heuristic	4	3.8	3.3
4	LSI & PCA	5	4.0	3.0	POS tagging & stopword removal	5	1.1	1.7	dependency parsing & logistic regression	6	4.8	3.0	Stanford Parser & dependency parsing	4	3.6	3.2
5	BoW & LSI	4	3.3	3.0	BoW & tokenization	4	1.1	1.8	LSI & PCA	6	3.6	2.5	BoW & disambiguation	4	3.1	2.9
6	dependency parsing & heuristic	5	3.3	2.7	alignment & tokenization	4	1.1	1.8	CoreNLP & stopword removal	5	3.4	2.7	steeming & stopword removal	7	2.8	2.1
7	dependency parsing & semantic features	4	1.8	0.5	PMI & POS tagging	4	0.9	1.8	lower case & steeming	4	3.2	3.0	IDF & syntactic features	4	2.8	2.7
8	POS tagging & lexical features	4	1.8	2.2	disambiguation & lemmatization	4	0.9	1.7	LSI & semantic features	11	3.0	1.8	CoreNLP & syntactic features	5	2.4	2.3
9	alignment & linear regression	5	1.6	1.9	POS tagging & n-grams	5	0.5	1.4	CoreNLP & lemmatization	5	2.8	2.5	disambiguation & lemmatization	6	2.3	2.1
10	IDF & n-grams	4	1.6	2.1	lemmatization & semantic features	4	0.2	1.3	lemmatization & linear regression	9	2.7	1.9	SVM & alignment	19	2.3	1.5
	2016				2017				2018				2019			
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α
1	LSTM & RNN	8	8.7	3.7	IDF & term frequency	10	10.9	3.7	scikit-learn & semantic features	5	3.6	2.9	BiLSTM & LSTM	4	1.6	2.1
2	BiLSTM & LSTM	5	8.4	5.5	FNN & fully connected layer	5	8.2	5.3	SVM & scikit-learn	4	1.8	2.2	POS tagging & dependency parsing	5	1.0	1.7
3	IDF & term frequency	11	8.2	2.8	fully connected layer & pooling	7	7.1	3.5	SVM & semantic features	5	1.6	2.0	LSTM & dependency parsing	4	0.5	1.5
4	CNN & pooling	5	8.0	5.2	CNN & pooling	8	6.7	3.0	logistic regression & semantic features	4	1.3	1.9	ELMo & tokenization	4	0.4	1.4
5	pooling & softmax	6	6.6	3.7	GRU & RNN	5	6.4	4.2	Word Embeddings & scikit-learn	6	0.9	1.6	LSTM & POS tagging	4	0.4	1.4
6	BiLSTM & RNN	4	6.5	5.3	GBM & scikit-learn	4	6.1	4.9	GloVe & SVM	6	0.9	1.6	POS tagging & Word Embeddings	6	0.2	1.2
7	Brown & clustering	6	6.5	3.6	steeming & stopword removal	7	5.9	3.1	Word Embeddings & normalization	5	0.9	1.6	Word Embeddings & tokenization	5	0.1	1.2
8	fully connected layer & softmax	5	5.3	3.6	lexical features & lower case	4	5.3	4.3	Word Embeddings & semantic features	8	0.9	1.5	ELMo & Word Embeddings	6	0.1	1.2
9	LSTM & softmax	7	4.7	2.7	LSTM & RNN	8	5.1	2.6	PMI & skipgram	4	0.9	1.7	BiLSTM & Word Embeddings	4	0.1	1.1
10	CNN & softmax	5	4.7	3.3	CNN & fully connected layer	6	4.8	2.9	GloVe & PMI	5	0.6	1.5	LSTM & Word Embeddings	5	0.1	1.1

Supplementary Figure 3: Semantics Analysis tasks in SemEval 2013-2019: Top 10 two-component collocations ranked by G^2 (only positive association: odds ratio $\alpha > 1$)

Rank	2012				2013				2014				2015			
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α
1	POS tagging, lemmatization, tokenization	0.186	50	16.0	POS tagging, lemmatization, stopword removal	0.192	27	14.5	POS tagging, lemmatization, semantic features	0.159	42	17.5	POS tagging, alignment, tokenization	0.378	173	3.1
2	alignment, normalization, tokenization	0.163	45	13.1	POS tagging, lemmatization, tokenization	0.231	27	4.4	LSI, semantic features, tokenization	0.143	40	16.1	POS tagging, SVM, alignment	0.356	166	4.1
3	normalization, stopword removal, tokenization	0.163	44	11.1	BoW, lemmatization, tokenization	0.115	16	8.8	LSI, SVM, semantic features	0.111	39	53.2	POS tagging, alignment, semantic features	0.356	162	2.9
4	POS tagging, lemmatization, normalization, stopword removal, tokenization	0.116	42	82.1	BoW, POS tagging, lemmatization, tokenization	0.115	16	8.8	n-grams, normalization, tokenization	0.159	38	9.5	POS tagging, semantic features, tokenization	0.356	162	2.8
5	alignment, semantic features, tokenization	0.163	42	8.5	POS tagging, lemmatization, semantic features, tokenization	0.115	16	9.9	lemmatization, semantic features, tokenization	0.143	32	7.7	POS tagging, alignment, lemmatization	0.333	159	6.0
6	lemmatization, semantic features, stopword removal	0.163	41	7.2	lemmatization, semantic features, tokenization	0.115	14	5.6	POS tagging, lemmatization, tokenization	0.159	30	5.4	alignment, semantic features, tokenization	0.333	155	4.3
7	alignment, normalization, semantic features	0.163	40	6.1	POS tagging, alignment, lemmatization, tokenization	0.115	14	5.6	POS tagging, lemmatization, semantic features, tokenization	0.111	30	12.6	POS tagging, SVM, tokenization	0.333	150	2.5
8	lemmatization, normalization, semantic features	0.163	40	6.1	alignment, lemmatization, tokenization	0.115	14	5.6	LSI, POS tagging, semantic features	0.111	29	10.3	POS tagging, lemmatization, semantic features	0.311	144	4.0
9	lemmatization, stopword removal, tokenization	0.163	39	6.3	PMI, POS tagging, tokenization	0.115	14	6.2	LSI, lemmatization, semantic features	0.111	29	10.3	SVM, alignment, tokenization	0.311	143	3.7
10	POS tagging, semantic features, stopword removal	0.163	39	7.1	BoW, POS tagging, tokenization	0.115	14	4.7	LSI, SVM, tokenization	0.111	28	9.8	POS tagging, alignment, normalization	0.311	141	2.8
2016																
Rank	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α
1	Word Embeddings, tokenization, w2v	0.234	95	5.3	Word Embeddings, tokenization, w2v	0.233	91	7.3	GloVe, SVM, Word Embeddings	0.3	92	13.2	POS tagging, Word Embeddings, dependency parsing	0.5	138.7	14.7
2	POS tagging, lemmatization, tokenization	0.221	92	6.1	Word Embeddings, n-grams, w2v	0.217	86	6.0	Word Embeddings, scikit-learn, semantic features	0.25	78	17.9	LSTM, POS tagging, Word Embeddings, dependency parsing	0.4	111.4	15.7
3	semantic features, tokenization, w2v	0.208	82	5.2	IDF, Word Embeddings, term frequency	0.15	82	65.7	Word Embeddings, skipgram, w2v	0.25	76	12.2	LSTM, Word Embeddings, dependency parsing	0.4	111.4	15.7
4	Word Embeddings, semantic features, tokenization	0.208	76	2.9	CNN, Word Embeddings, pooling	0.133	77	118.4	SVM, Word Embeddings, semantic features	0.25	75	8.2	LSTM, POS tagging, dependency parsing	0.4	111.4	15.7
5	alignment, semantic features, tokenization	0.208	74	2.5	LSTM, RNN, Word Embeddings	0.133	76	90.5	GloVe, Word Embeddings, w2v	0.250	70	2.3	LSTM, POS tagging, Word Embeddings	0.4	108.8	5.9
6	Word Embeddings, alignment, tokenization	0.182	70	4.2	Word Embeddings, semantic features, tokenization	0.2	73	3.0	GloVe, PMI, skipgram	0.2	64	25.2	ELMo, Word Embeddings, tokenization	0.4	108.8	5.9
7	POS tagging, alignment, tokenization	0.195	69	2.5	Word Embeddings, n-grams, tokenization	0.2	73	3.0	SVM, Word Embeddings, scikit-learn, semantic features	0.2	64	25.2	BiLSTM, LSTM, dependency parsing	0.300	83.78	14.7
8	alignment, tokenization, w2v	0.182	67	3.1	POS tagging, Word Embeddings, alignment	0.183	70	4.2	SVM, scikit-learn, semantic features	0.2	64	25.2	BiLSTM, LSTM, Word Embeddings, dependency parsing	0.3	83.78	14.7
9	stopword removal, tokenization, w2v	0.156	64	5.6	Word Embeddings, fully connected layer, pooling	0.117	67	80.8	RF, Word Embeddings, w2v	0.2	61	10.7	BiLSTM, LSTM, POS tagging	0.3	83.78	14.7
10	SVM, alignment, w2v	0.143	63	10.1	n-grams, tokenization, w2v	0.167	66	5.4	SVM, Word Embeddings, scikit-learn	0.2	60	7.2	BiLSTM, LSTM, POS tagging, Word Embeddings, dependency parsing	0.3	83.78	14.7

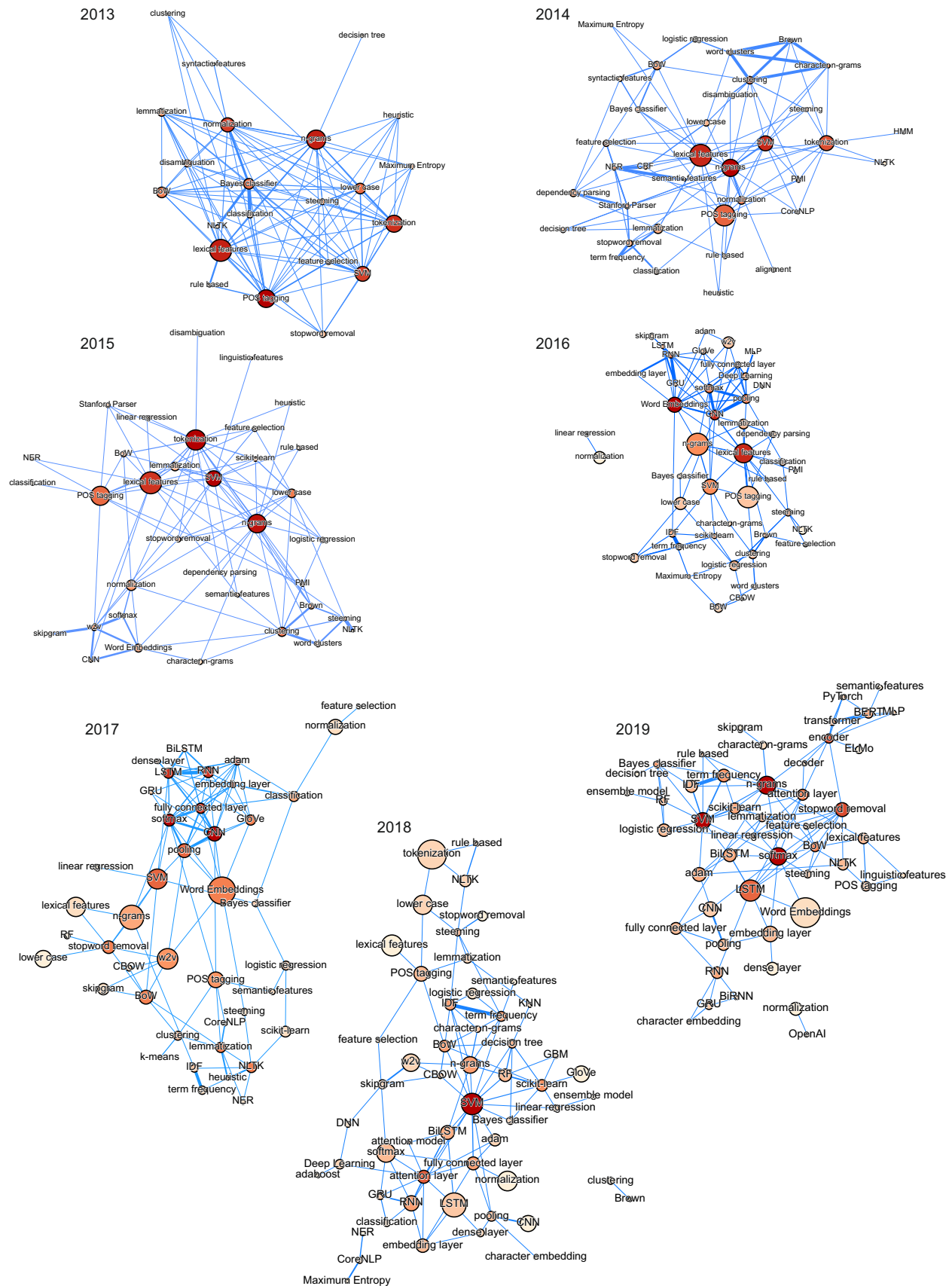
Supplementary Figure 4: Semantics Analysis tasks in SemEval 2013-2019: Top 10 collocations of ≥ 3 -components derived from rules found using Apriori algorithm, ranked by G^2 (only positive association: odds ratio $\alpha > 1$)

Rank	2013				2014				2015							
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α				
1	lexical features & semantic features	4	4.6	4.0	disambiguation & word sense disambiguation	4	3.3	3.2	SRL & semantic features	4	4.4	3.8				
2	CRF & NER	4	3.4	3.1	POS tagging & word sense disambiguation	4	2.3	2.5	CRF & NER	7	3.7	2.4				
3	SVM & steeming	4	2.2	2.5	POS tagging & disambiguation	8	0.9	1.5	alignment & lemmatization	5	2.8	2.5				
4	alignment & lemmatization	5	1.3	1.8	disambiguation & lexical features	4	0.7	1.6	lemmatization & semantic features	7	2.2	1.9				
5	ME & normalization	4	1.0	1.8	CRF & tokenization	11	0.6	1.3	CRF & normalization	8	2.2	1.8				
6	NER & tokenization	7	1.0	1.5	disambiguation & normalization	7	0.6	1.4	normalization & rule based	4	1.9	2.3				
7	SVM & n-grams	5	0.9	1.6	POS tagging & tokenization	10	0.5	1.3	disambiguation & normalization	4	1.9	2.3				
8	POS tagging & lemmatization	13	0.8	1.3	POS tagging & alignment	5	0.4	1.4	alignment & semantic features	4	1.6	2.1				
9	CRF & tokenization	5	0.7	1.6	POS tagging & normalization	8	0.4	1.3	SVM & rule based	4	1.2	1.9				
10	CRF & normalization	4	0.7	1.6	lexical features & normalization	4	0.4	1.4	POS tagging & rule based	5	1.1	1.7				
	2016				2017				2018				2019			
	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α	Collocation	Freq	G ²	α
1	SVM & Word Embeddings	4	0.6	1.6	BiLSTM & LSTM	4	6.1	5.0	attention layer & attention model	5	6.9	4.5	IDF & term frequency	7	7.5	3.7
2	POS tagging & tokenization	10	0.3	1.2	CNN & pooling	4	5.6	4.6	decoder & encoder	4	6.7	5.5	BERT & transformer	9	6.9	2.9
3	lower case & tokenization	6	0.3	1.3	fully connected layers & pooling	4	5.3	4.4	IDF & term frequency	5	6.3	4.1	BERT & encoder	9	5.6	2.6
4	lexical features & tokenization	4	0.3	1.3	fully connected layers & softmax	6	5.3	3.2	lower case & normalization	4	6.0	4.9	ELMo & fastText	4	5.3	4.3
5	POS tagging & lexical features	4	0.2	1.3	LSTM & softmax	5	4.8	3.4	decision tree & lower case	4	5.3	4.3	BiLSTM & LSTM	10	5.1	2.3
6	dependency parsing & tokenization	4	0.2	1.3	CRF & NER	7	4.7	2.7	Scikit learn & n-grams	4	5.0	4.1	POS tagging & lexical features	4	4.5	3.8
7	CRF & tokenization	7	0.2	1.2	FNN & softmax	6	4.6	2.9	Bayes classifier & SVM	4	5.0	4.1	BiLSTM & RNN	6	4.4	2.8
8	POS tagging & lower case	6	0.2	1.2	CNN & fully connected layers	5	4.5	3.2	NLTK & lower case	4	5.0	4.1	dense layer & fully connected layers	7	4.1	2.5
9	POS tagging & Word Embeddings	6	0.1	1.2	pooling & softmax	4	4.4	3.7	attention layer & encoder	5	4.6	3.2	SVM & term frequency	6	3.9	2.6
10	POS tagging & dependency parsing	4	0.1	1.2	NER & lemmatization	6	4.3	2.8	GloVe & encoder	6	4.3	2.8	encoder & ensemble model	6	3.8	2.6

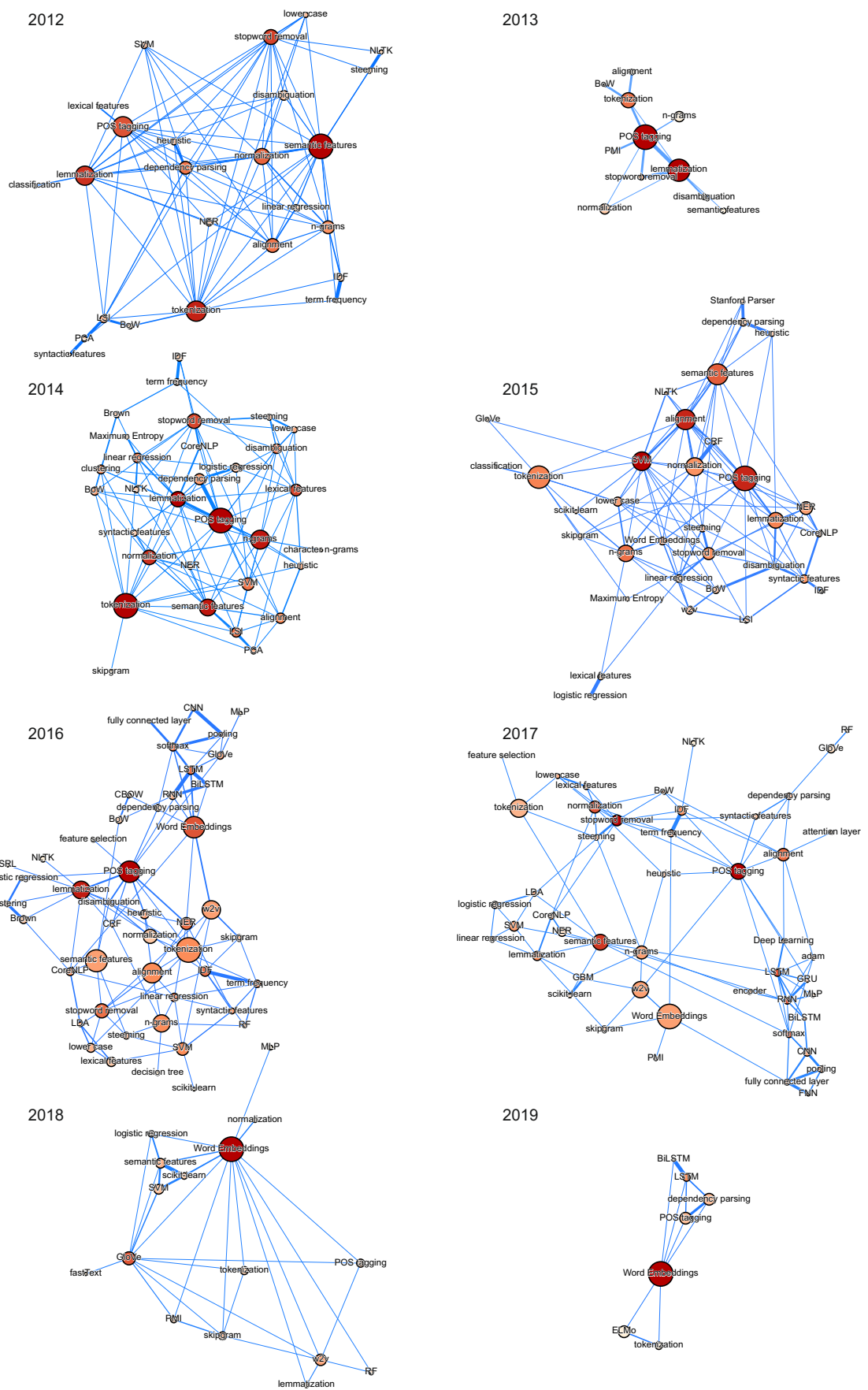
Supplementary Figure 5: Information Extraction tasks in SemEval 2013-2019: Top 10 two-component collocations ranked by G^2 (only positive association: odds ratio $\alpha > 1$)

Rank	2013				2014				2015							
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α				
1	POS tagging, lemmatization, tokenization	0.476	87	3.3	CRF, POS tagging, tokenization	0.692	141	4.4	POS tagging, n-grams, tokenization	0.31	95	6.3				
2	POS tagging, normalization, tokenization	0.429	79	5.0	POS tagging, normalization, tokenization	0.615	126	7.8	CRF, POS tagging, tokenization	0.276	84	5.3				
3	POS tagging, lemmatization, normalization	0.381	70	4.7	CRF, normalization, tokenization	0.615	123	2.6	CRF, normalization, tokenization	0.241	83	28.7				
4	POS tagging, lemmatization, normalization, tokenization	0.333	63	6.1	disambiguation, normalization, tokenization	0.538	111	10.3	POS tagging, SVM, n-grams	0.276	83	5.2				
5	lemmatization, normalization, tokenization	0.333	61	4.0	CRF, POS tagging, normalization, tokenization	0.538	108	4.6	POS tagging, SVM, normalization	0.241	76	8.0				
6	POS tagging, n-grams, tokenization	0.286	48	1.0	CRF, POS tagging, normalization	0.538	107	3.5	CRF, POS tagging, normalization, tokenization	0.207	73	27.7				
7	POS tagging, lemmatization, n-grams	0.286	47	1.4	CRF, disambiguation, tokenization	0.538	106	1.7	POS tagging, dependency parsing, tokenization	0.241	72	4.2				
8	POS tagging, lemmatization, rule based, tokenization	0.238	47	9.8	POS tagging, disambiguation, tokenization	0.538	106	1.7	POS tagging, normalization, tokenization	0.241	72	4.2				
9	NER, normalization, tokenization	0.238	44	4.4	POS tagging, disambiguation, normalization, tokenization	0.462	93	5.9	POS tagging, SVM, tokenization	0.241	72	4.2				
10	lemmatization, rule based, tokenization	0.238	44	4.4	CRF, disambiguation, normalization	0.462	93	5.9	CRF, NER, tokenization	0.207	68	11.1				
	2016				2017				2018				2019			
	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α	Collocation	supp	G ²	α
1	CRF, POS tagging, tokenization	0.462	75	1.2	POS tagging, Word Embeddings, tokenization	0.4	149	6.3	Word Embeddings, fully connected layers, softmax	0.4	251	26.5	FNN, Word Embeddings, softmax	0.4	227.3	20.8
2	POS tagging, Word Embeddings, tokenization	0.385	62	2.0	CRF, POS tagging, tokenization	0.333	122	4.5	FNN, Word Embeddings, w2v	0.375	221	6.0	LSTM, Word Embeddings, softmax	0.375	200.6	7.2
3	POS tagging, SVM, Word Embeddings	0.308	54	11.8	POS tagging, n-grams, tokenization	0.3	108	2.9	FNN, Word Embeddings, softmax	0.375	216	3.4	GloVe, Word Embeddings, softmax	0.325	174.1	5.5
4	CRF, POS tagging, lower case, tokenization	0.308	52	4.9	POS tagging, lower case, tokenization	0.3	108	2.9	Word Embeddings, softmax, tokenization	0.35	206	5.5	FNN, LSTM, Word Embeddings, softmax	0.3	172.2	27.3
5	CRF, lower case, tokenization	0.308	50	2.6	POS tagging, Word Embeddings, w2v	0.233	90	8.6	CNN, FNN, Word Embeddings	0.350	205	5.0	FNN, LSTM, softmax	0.3	168.8	19.5
6	CRF, POS tagging, lower case	0.308	50	2.6	FNN, Word Embeddings, softmax	0.2	88	60.3	Word Embeddings, tokenization, w2v	0.325	191	5.3	FNN, LSTM, Word Embeddings	0.325	168.5	3.1
7	POS tagging, SVM, tokenization	0.308	49	1.3	Word Embeddings, fully connected layers, softmax	0.2	88	60.3	CNN, Word Embeddings, softmax	0.325	189	4.5	CNN, Word Embeddings, softmax	0.300	164.4	9.4
8	POS tagging, SVM, Word Embeddings, tokenization	0.231	39	5.9	CRF, lower case, tokenization	0.233	87	4.8	FNN, LSTM, Word Embeddings	0.325	188	4.6	GloVe, LSTM, Word Embeddings	0.3	154.6	2.9
9	SVM, Word Embeddings, tokenization	0.231	39	5.9	POS tagging, SVM, tokenization	0.233	85	4.7	FNN, fully connected layers, softmax	0.3	186	23.0	CNN, FNN, softmax	0.275	153.5	16.3
10	CRF, POS tagging, lemmatization, tokenization	0.231	39	5.9	NER, POS tagging, tokenization	0.233	83	2.5	Word Embeddings, embedding layer, softmax	0.3	186	23.0	Word Embeddings, fully connected layers, softmax	0.275	152.2	10.7

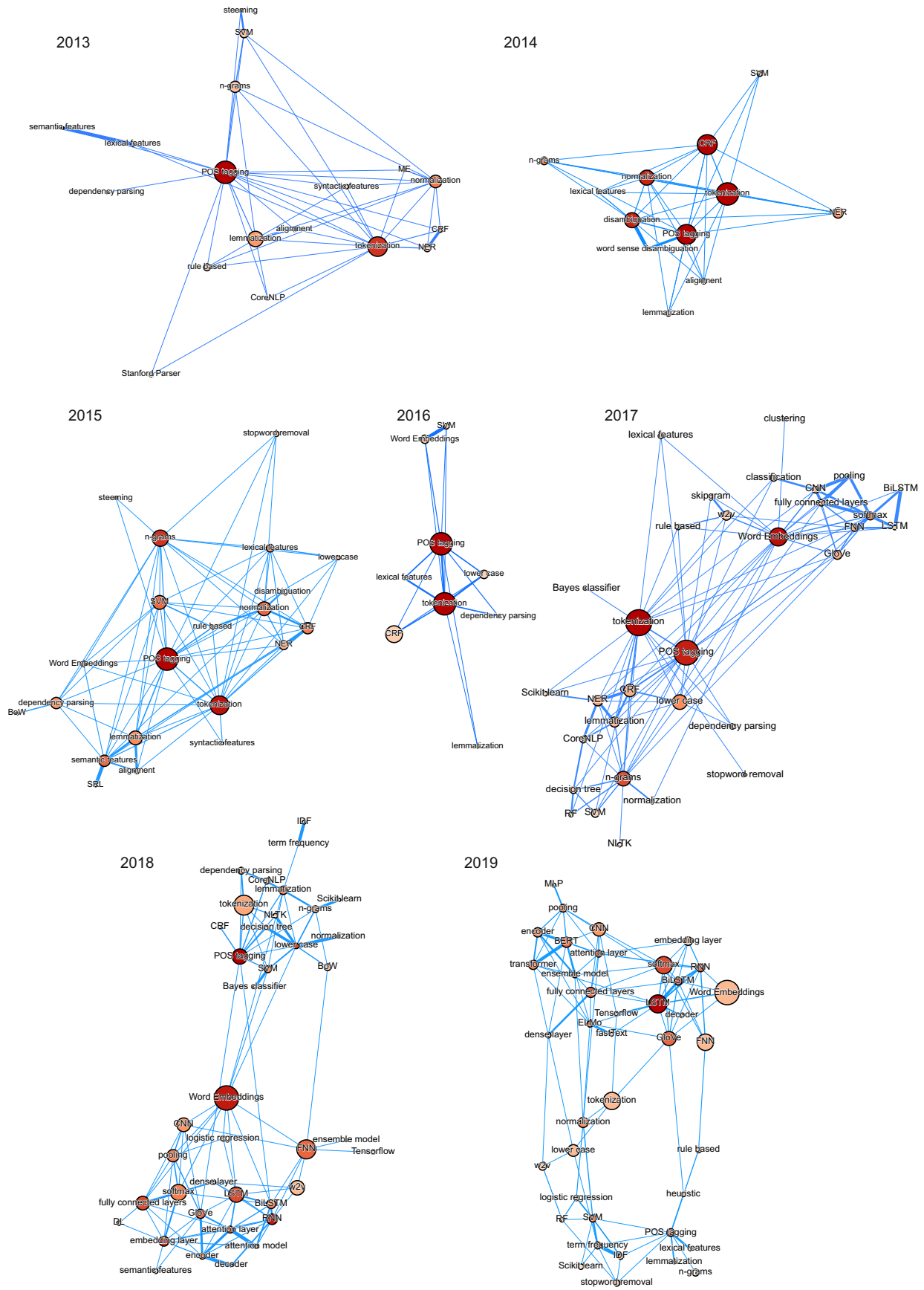
Supplementary Figure 6: Information Extraction tasks in SemEval 2013-2019: Top 10 collocations of ≥ 3 -components derived from rules found using Apriori algorithm, ranked by G^2 (only positive association: odds ratio $\alpha > 1$)



Supplementary Figure 7: Network graphs built using the top 100 collocations from SA tasks in SemEval from 2013 to 2019. Node size is proportional to the frequency of the component, the red color intensity to node degree, edge width to G^2 of the collocation. For full resolution see online version of the paper.



Supplementary Figure 8: Network graphs built using the top 100 collocations from SEM tasks in SemEval from 2012 to 2019. Node size is proportional to the frequency of the component, the red color intensity to node degree, edge width to G^2 of the collocation. For full resolution see online version of the paper.



Supplementary Figure 9: Network graphs built using the top 100 collocations from IE tasks in SemEval from 2013 to 2019. Node size is proportional to the frequency of the component, the red color intensity to node degree, edge width to G^2 of the collocation. For full resolution see online version of the paper.