

Basics of subproject 4.6 – Plato's aftereffects

Text Mining for Classical Studies
Leipzig, 2009/10/02

Marco Büchler
Natural Language Processing Group
Department of Computer Science
University of Leipzig

- Research question
- Approaches from NLP (Natural Language Processing)
 - Multi word expression/unit detection
 - Positional inverted lists
 - Categorized term extraction (examples)
- Main objectives
- Challenges

Platon's aftereffects: Research question

- Which aftereffects of Plato's works can be observed?
 - Which methods of multi word /phrase extraction can be used?
- What is a citation? / What is a phrase?
- More specific:
 - What is a citation/phrase in a language with a free word order?
 - English example of a phrase:

As soon as possible
 - At least 3 of 4 words are stop words
 - What would this mean with free word order?

Plato's aftereffects: Approaches from NLP

- 1. Step:
 - Find word by word citations!
 - *Iterated neighbourhood co-occurrences*

- 2. Step:
 - Find citations allowing free word order
 - *Positional inverted list combined with distance filters and similarity measures*

- 3. Step (experimental):
 - Semantic search of citations
 - *Similarity of co-occurrence profiles*

- Step 1: Compute significant neighbourhood co-occurrences
- Step 2: Handle results of step before as one word and observe new (more complex) significant neighbourhood co-occurrences
- Step 3: Removing prefixes of a phrase if:

$$eps_{FC} \leq \log\left(\frac{P(\text{word}_1, \text{word}_2 \dots \text{word}_n)}{P(\text{word}_1, \text{word}_2 \dots \text{word}_n, \text{word}_{n+1})}\right)$$

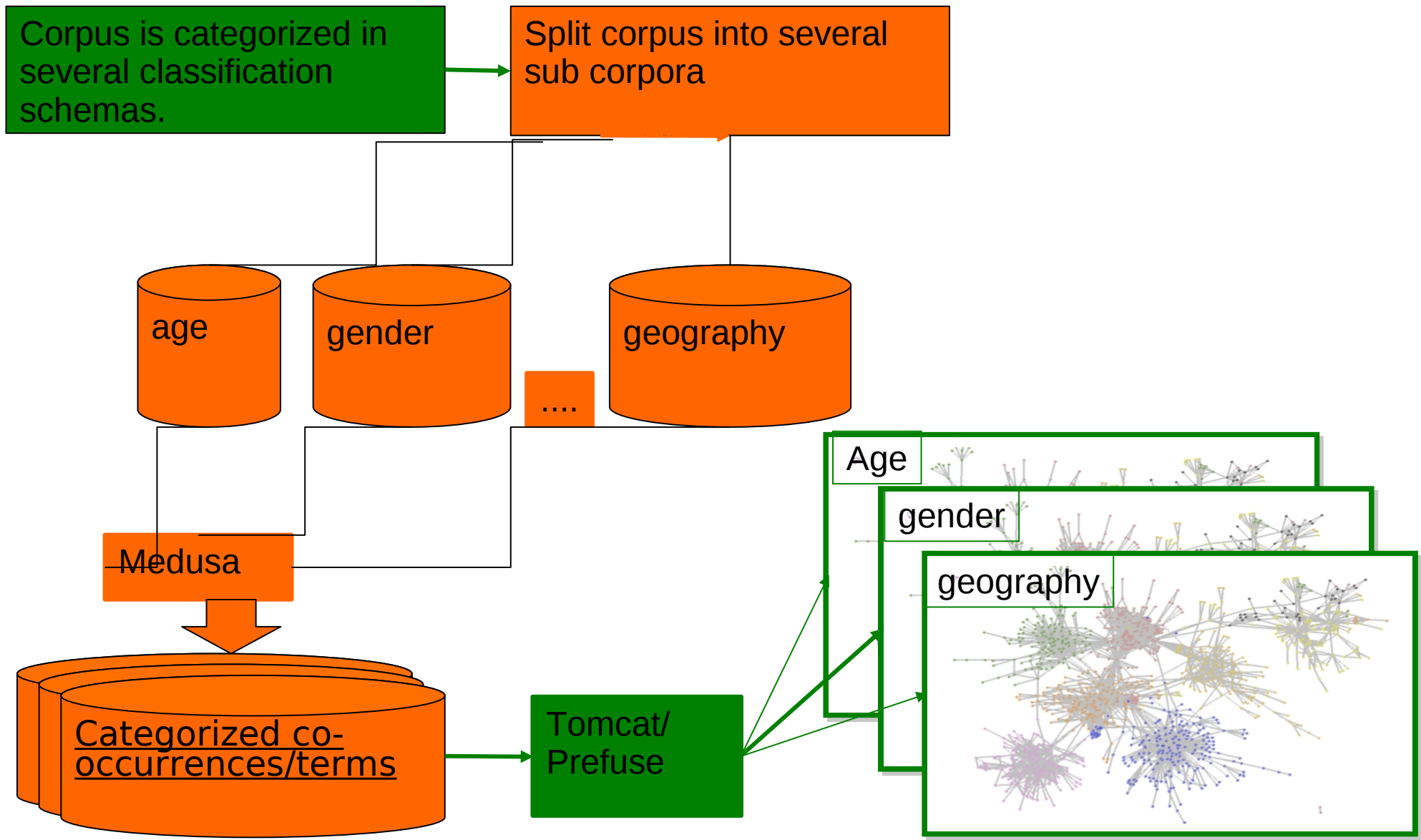
- Step 4: Iterating step 2-4 until
 - Maximum of loops are reached or
 - Nothing can be extracted
- Configuration parameters:
 - Number of loops (step 4): Default value 50
 - Statistical significance threshold (step 1/2): Default value 3.84
 - Minimum frequency (step 1/2): Default value 3 (currently 2)
 - Frequency ratio (step 3): Default value 1.0

- First results based on TLG:
 - 6,296,593 possible phrases/citations
 - Algorithm delay: about 3 weeks
- An example (eAQUA website – Numbers are arbitrarily used in this example):
 - *The eAqua-project aims at generating specific knowledge from ancient texts and will provide this knowledge via an open web-portal to the scientific community for future empirical studies.*
 - Step 1: Computing significant neighbourhood co-occurrences
 - e. g. **(The, eAqua-project, 36.7, 6)**, *(generating, specific, 54.1, 7)*, **(specific, knowledge, 109.1, 10)**,
 - Step 2: Computing significant multi word units and words
 - e. g. **(generating specific, knowledge, 34.1, 5)**, **(specific knowledge, from, 109.1, 3)**,
 - Step 3: Remove prefixes of possible phrases
 - e. g. **(specific knowledge, from, 109.1, 3)**,

- Problem:
 - Longer phrases can be detected however it's difficult to handle short phrases (Typical case: syntactical fragments)
- Possible solutions:
 - Using author's name as fix points (like Plato said)
 - Using significant terms of Plato or his works

Work	Phrases selected by number words																										
	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	
<i>Alcibiades_i</i>	3771	2076	809	344	186	102	59	39	26	16	10	7	4	2	1	0	0	0	0	0	0	0	0	0	0	0	
<i>Alcibiades_ii</i>	1568	691	124	33	10	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Amatores</i>	906	309	46	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Apologia_Socratis</i>	3189	1531	539	290	190	135	100	78	62	51	43	35	27	21	15	9	5	2	0	0	0	0	0	0	0	0	
<i>Charmides</i>	2709	1082	221	49	13	8	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Clitophon</i>	679	268	66	19	10	7	5	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Cratylus</i>	5189	2646	554	129	56	33	21	12	7	5	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Critias</i>	1665	767	150	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Crito</i>	1642	794	249	113	68	51	36	26	18	13	10	7	4	2	1	0	0	0	0	0	0	0	0	0	0	0	
<i>Definitiones</i>	498	195	47	9	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Epigrammata</i>	478	293	179	107	59	29	13	7	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Epinomis</i>	2224	935	214	62	36	22	14	10	7	5	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Epistulae</i>	5337	2624	693	255	152	99	68	48	33	24	15	9	6	4	2	1	0	0	0	0	0	0	0	0	0	0	
<i>Euthydemus</i>	3619	1497	245	34	12	8	6	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Euthyphro</i>	1819	787	146	21	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Gorgias</i>	8176	4738	1550	653	369	221	142	92	58	36	23	15	7	4	2	1	0	0	0	0	0	0	0	0	0	0	
<i>Hipparchus</i>	861	296	48	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Hippias_major</i>	2786	1224	242	37	12	5	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Hippias_minor</i>	1565	646	169	62	41	23	13	9	7	5	4	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Ion</i>	1429	617	151	53	26	12	6	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Laches</i>	2553	1045	177	32	11	7	5	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Leges</i>	22385	13109	3933	1588	989	665	467	333	230	164	116	81	58	41	31	24	18	12	8	6	4	3	2	1	0	0	
<i>Lysis</i>	2212	885	147	34	14	9	5	4	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Menexenus</i>	1849	946	323	149	98	67	50	38	27	19	15	11	8	6	5	4	3	2	1	0	0	0	0	0	0	0	
<i>Meno</i>	3279	1399	281	52	14	7	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Minos</i>	1123	437	72	12	4	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Parmenides</i>	4261	2974	1089	460	275	176	112	70	47	32	22	14	10	6	3	2	1	0	0	0	0	0	0	0	0	0	
<i>Phaedo</i>	7413	5038	2376	1423	1009	727	522	363	244	162	109	77	54	38	27	18	10	6	3	1	0	0	0	0	0	0	
<i>Phaedrus</i>	5698	2944	1005	457	274	171	109	66	45	29	18	12	7	4	3	2	1	0	0	0	0	0	0	0	0	0	
<i>Philebus</i>	5094	2560	688	259	149	102	68	46	29	16	7	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Politicus</i>	4944	2301	530	147	76	41	23	15	9	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Protagoras</i>	5312	2464	551	123	51	26	15	9	6	4	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Respublica</i>	20658	12827	4069	1710	1053	711	481	318	218	154	106	73	48	32	19	10	6	4	3	2	1	0	0	0	0	0	
<i>Sophista</i>	4782	2604	847	386	266	192	140	100	72	52	39	29	20	13	8	5	3	1	0	0	0	0	0	0	0	0	
<i>Spuria</i>	4515	2117	467	87	17	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Symposium</i>	5505	2624	617	133	45	23	12	6	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Theaetetus</i>	6852	3637	1215	627	425	304	221	158	110	75	48	30	19	10	5	3	2	1	0	0	0	0	0	0	0	0	
<i>Theages</i>	1274	503	91	14	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<i>Timaeus</i>	8338	6765	4202	2812	1993	1437	1035	750	539	388	284	209	147	106	81	63	49	37	27	20	14	9	6	4	2	1	

	TLG v0.1			TLG v0.2			TLG v0.3		
number of words	155			26			13		
sum of frequencies:	16312			16092			16093		
	Rank	Word	Freq.	Rank	Word	Freq.	Rank	Word	Freq.
	741	Πλάτων	8059	692	Πλάτων	8730	689	Πλάτων	8758
	1555	Πλάτωνος	3495	1554	Πλάτωνος	3813	1556	Πλάτωνος	3813
	3122	Πλάτωνα	1671	3220	Πλάτωνα	1803	3226	Πλάτωνα	1804
	3612	Πλάτωνι	1418	3794	Πλάτωνι	1534	3786	Πλάτωνι	1538
	22617	Πλάτων·	205	75374	Πλάτωνός	55	75460	Πλάτωνός	55
	31711	Πλατωνικῶν	141	85272	Πλάτωνά	47	85351	Πλάτωνά	47
	49238	Πλάτωνος·	84	141061	Πλάτωνος	24	141115	Πλάτωνος	24
	53525	ΠΛΑΤΩΝΟΣ	76	145464	Πλάτωνί	23	145498	Πλάτωνί	23
	62353	Πλάτωνα·	63	166139	_Πλάτων	19	187888	Πλάτωνες	16
	63178	Πλατωνικὸν	62	187866	Πλάτωνες	16	241916	Πλάτωνας	11
	63986	Πλατωνικοὶ	61	241920	Πλάτωνας	11	695698	Πλάτωνο	2
	69635	Πλατωνικῆς	55	699357	Πλάτων_καὶ	2	1044813	Πλάτωνε	1
	71604	Πλατωνικὸς	53	699358	Πλάτωνο	2	1044814	Πλάτωνάς	1
	73838	ΠΛΑΤΩΝ	51	1066134	Πλάτων__	1			
	75004	Πλάτωνός	50	1066135	Πλάτων_βούλεται	1			
	77550	Πλάτωνι·	48	1066136	Πλάτων_τὸν	1			
	78821	Πλατωνικοῦ	47	1066137	Πλάτων_έν	1			



(Source: Taken from bachelor thesis slides of Marcus Puchalla.)

- Plato_Phil:

	Wort	Bedeutung	Relevant
1	ΘΕΑΙ	Abkürzung	+
2	ΞΕ	Abkürzung	+
3	ὦ	Ausruf (oh (Sokrates)!)	++
4	Σώκρατες	Personenname	+
5	ΑΘ	Abkürzung	+
6	ΠΡΩ	Abkürzung	+
7	Πάνυ	Adverb	+
8	ἔφη	Verb	+
9	ΚΛ	Abkürzung	++
10	_ΣΩ	Abkürzung	+
11	Οὐκοῦν	Partikel	-

(Source: Taken from bachelor thesis slides of Marcus Puchalla.)

- Data: TLG Ancient Greek corpus
- Algorithm: Positional inverted list
 - Example sentence: This is a presentation.

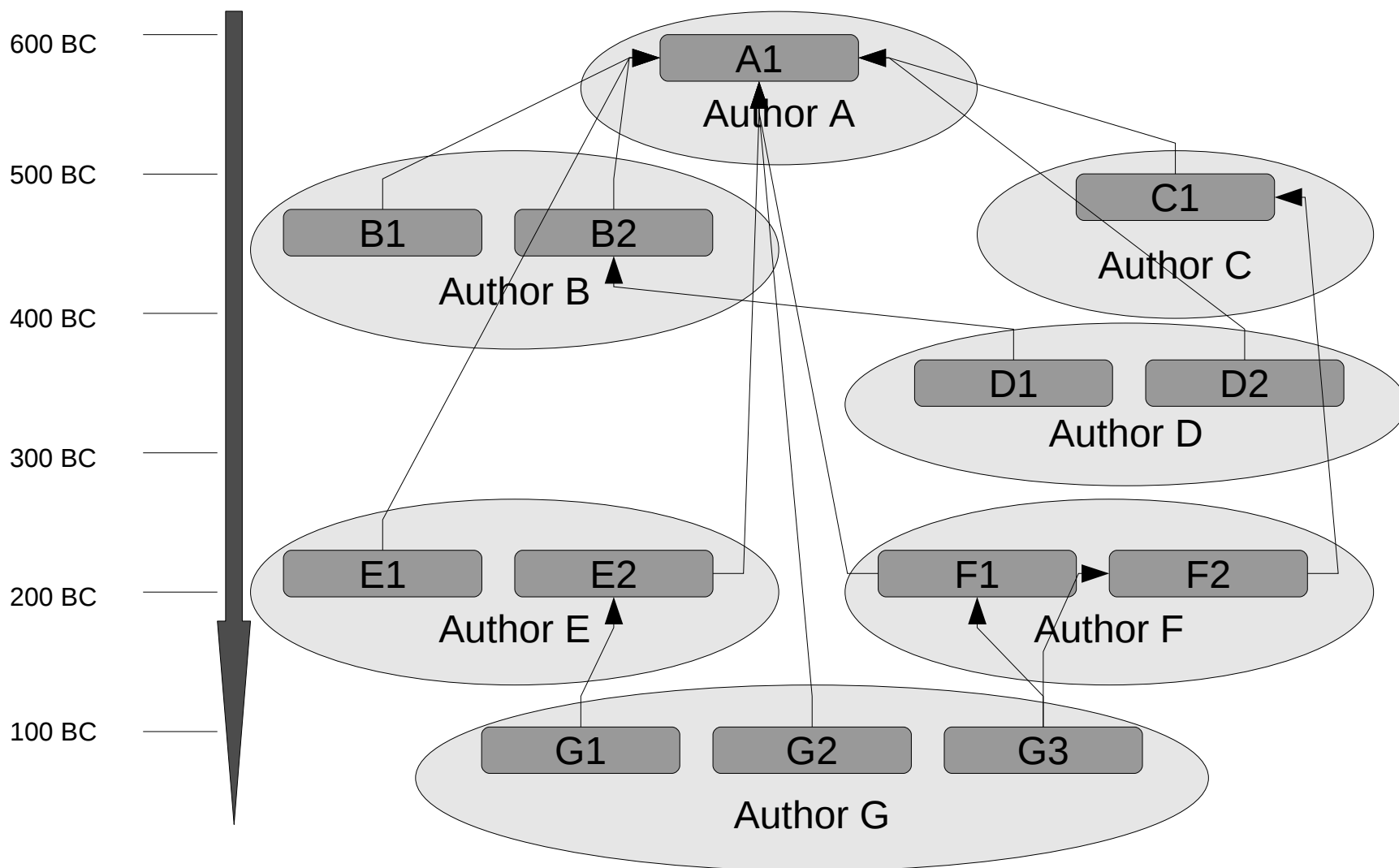
Word	Source	Position
This	1	0
is	1	1
a	1	2
presentation	1	3
.	1	4

- Based on TLG: 87,045,149 entries in database
- Application: Distance based concordances
 - Manually:
 - Requesting a set of words
 - Detection of all sentences (including their environment) matching the requested words
 - Compute the minimum and maximum positions of these words
 - Select all sentences containing too much non requested words between min and max.

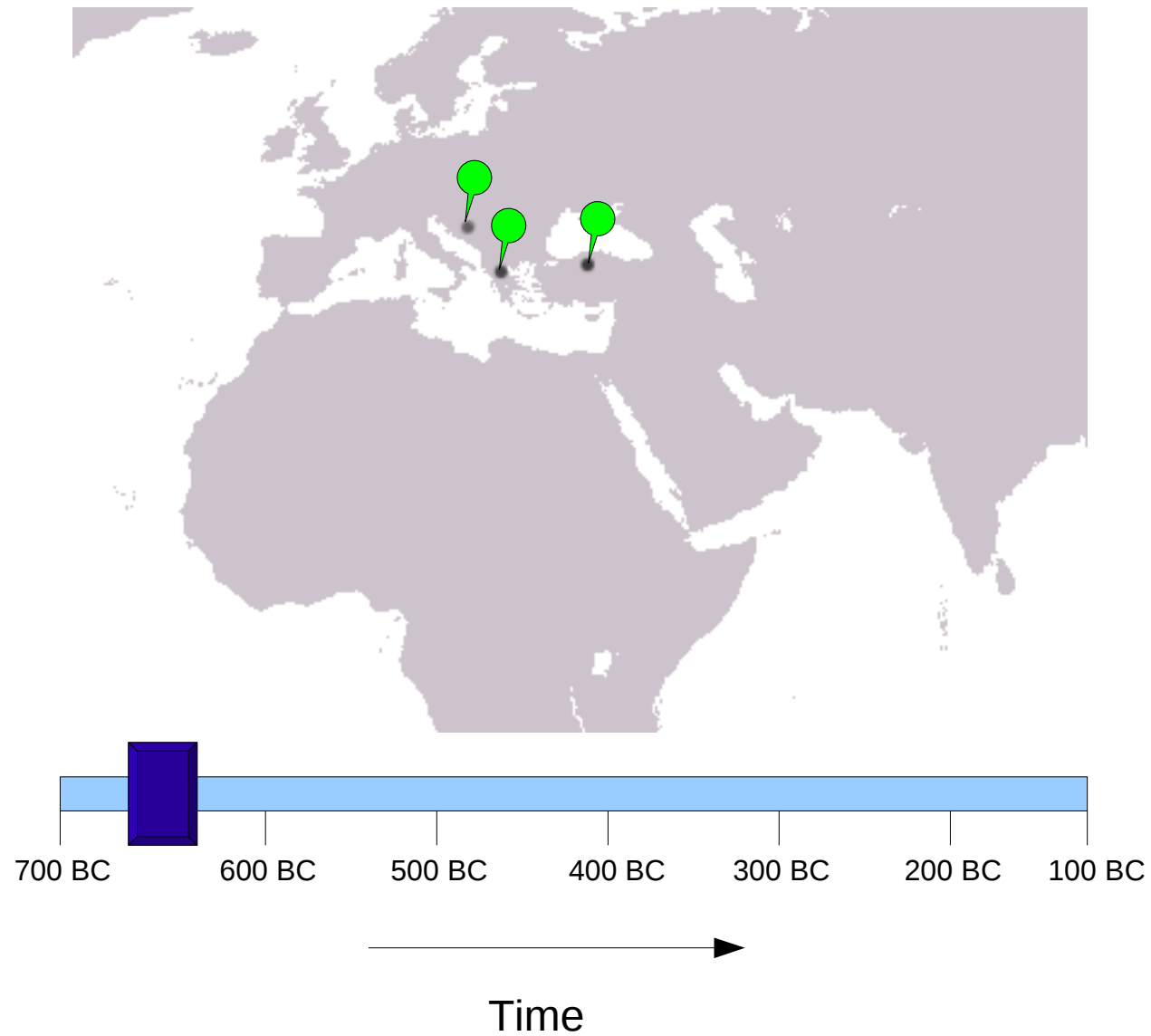
- Application: Distance based concordances
 - Automatically:
 - Naive method: comparing every sentence with all other sentences
 - TLG: $5,500,000 * 5,500,000 = 3.025e13$ comparisons
 - Assumption: It can be compared 1000 sentences/sec.
 - This process would run about **3.025e10 seconds** or more than **959 years**.
 - Even if we would compare only sentences with all significant phrases we would need about one year.
 - That's why:
 - Usage of divide & conquer strategies
 - Intelligent pre-clustering of data
 - Using occurrences of Plato, work titles or roles of Plato's works
 - Using significant terms of Plato's work

- Basic assumption:
 - Two words being semantically similar have the same co-occurrences
- Example:
 - laptop: {mouse, battery, display, portable}
 - computer: {mouse, keyboard, display, mainframe}
- Algorithm (simplification):
 - Compute co-occurrences for all words
 - Compare co-occurrence profiles of two words
 - Compute intersection **I** of co-occurrence profiles of both words
 - Compute union **U** of co-occurrence profiles of both words
 - Compute ratio $\text{sim} = \text{card}(\mathbf{I}) / \text{card}(\mathbf{U})$
- Example (continued):
 - $\mathbf{I} = \{\text{mouse, display}\}$
 - $\mathbf{U} = \{\text{mouse, battery, display, portable, keyboard, mainframe}\}$
 - $\text{sim} = \text{card}(\mathbf{I}) / \text{card}(\mathbf{U}) = 2/6 = 1/3$

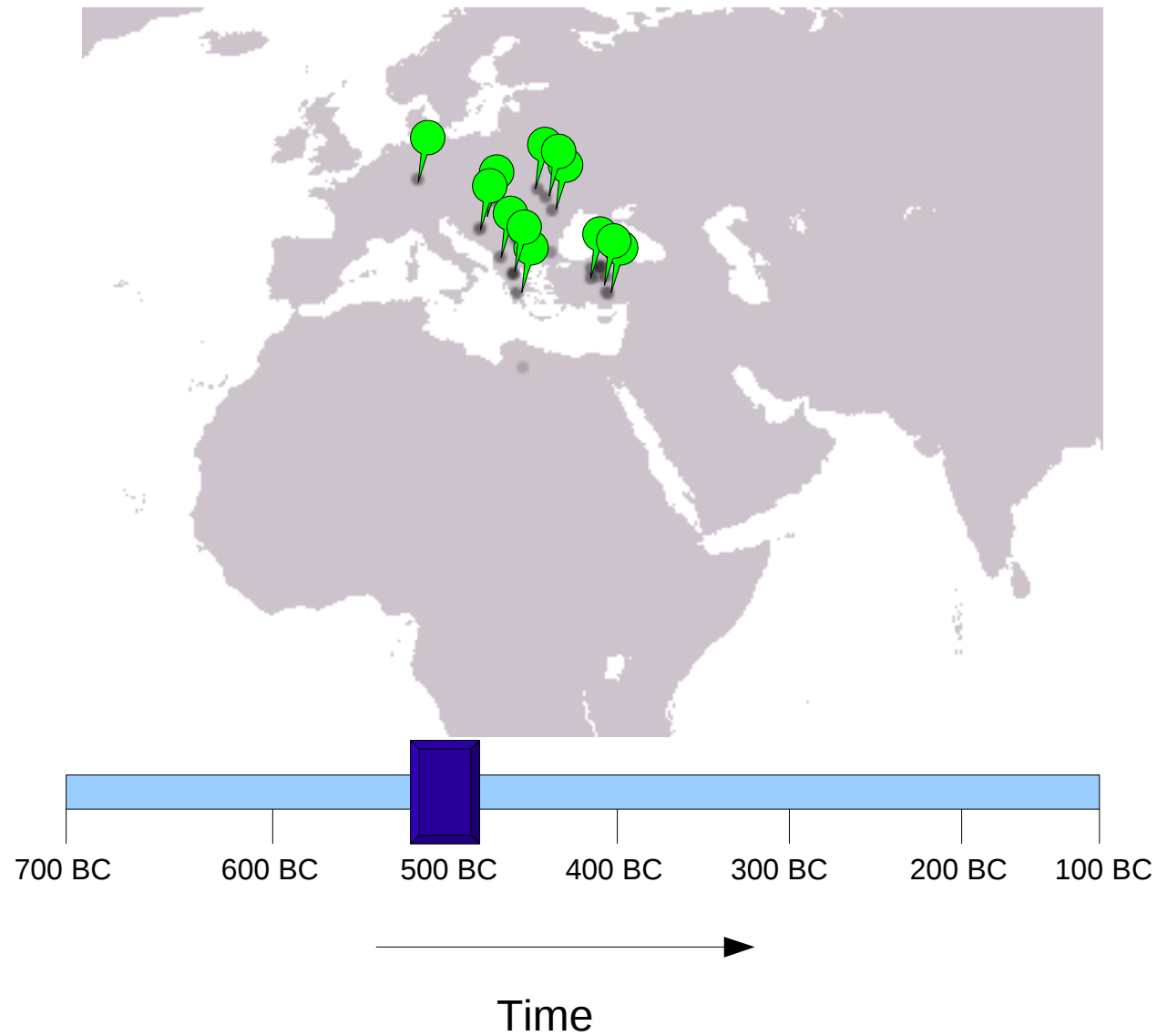
- A little bit more complex:
 - Removing $\text{card}(\mathbf{I})/\text{card}(\mathbf{U})$ by measuring the cosine value (angle) of two word's co-occurrence vectors



(Source: Taken from work of F. Cheema, G. Scheuermann)



(Source: Taken from work of F. Cheema, G. Scheuermann)



(Source: Taken from work of F. Cheema, G. Scheuermann)

Plato's aftereffects: Main objectives

- Computation of word by word phrases
- Computation of free word order phrases
- Definition/selection of citation from those computed phrases
- Recognition of Plato' citations in TLG.
- Expanding search by similar used words
- Good visualisation of citations

Plato's aftereffects: Challenges

- Degree of normalisation
- Theoretical complexity of algorithms
- phrase/citation vs. free word order
- Text preprocessing
 - Encoding problems etc.
 - How to handle words like: , . ? :