

Artificial Intelligence Conferences Closeness

Sébastien Konieczny and Emmanuel Lonca

CRIL - CNRS, Université d'Artois, France

{konieczny, lonca}@cril.fr

Abstract

We study the evolution of Artificial Intelligence conference closeness, using the **coscinus** tool. **coscinus** computes the closeness between publication supports using the co-publication habits of authors: the more authors publish in two conferences, the closer these two conferences. In this paper we perform an analysis of the main Artificial Intelligence conferences based on principal components analysis and clustering performed on this closeness relation.

1 Introduction

It is often hard and artificial to put boundaries on scientific domains. This is especially true for Artificial Intelligence, due to the highly different theoretical techniques that are used, and the huge spectrum of applications domains.

Similarly it can be difficult to judge the proximity or closeness of different Artificial Intelligence communities, and the historical evolution of this closeness.

One possibility is to look at the contents of the main conferences of these communities, in order to analyse this kind of change. This can be done for instance by trying to analyse and to match the title of the sessions of these conferences. But this process would require (subjective) interpretations and text analysis.

A more objective way of analysing the closeness between communities is by looking at the publication habits of researchers. Starting from the hypothesis that mosts researchers are specialists of a given single domain, or of a set of connected domains, this means that the publication supports (conferences and journals) used by a researcher connect these publications supports together (around this researcher expertise). So this means that if there are many researchers that publish papers in two distinct conferences, these conferences are clearly thematically connected. So, by looking at the publication habits of researchers, one can draw a landscape of publication supports, and then obtain objective information about the communities and the domains.

This is the aim of the **coscinus** tool (<http://www.coscinus.org>), that performs such an analysis from publications data issued from DBLP (<https://dblp.uni-trier.de>), that is currently the largest and most stable database of computer science

publications, with more than 4 millions of records (more than 1.5 millions of journal papers and more than 2 millions of conference and workshop papers) in 2018 (source - <https://dblp.org/statistics/recordsindblp.html>).

In order to perform a study of the historical evolution of Artificial Intelligence conference closeness, we have adapted this tool by adding a temporal window selection, and focusing on Artificial Intelligence publication supports. This adapted tool can be find here: <http://www.coscinus.org/ai>. In this paper we will sum up and analyze some observations that can be made by using this tool.

In Section 2 we will describe the **coscinus** methodology, how the closeness measure is defined, and how it is used for the principal component analysis and for the clustering. In Section 3 we will study the closeness of the IJCAI conference with conferences from other domains, in order to have a global view of this closeness with other main computer science conferences. In Section 4 we will perform a similar analysis but focusing only on IJCAI-related AI conferences.

2 Coscinus

The data used in the **coscinus** project (<http://www.coscinus.org>) are issued from two sources:

- The DBLP database (<https://dblp.uni-trier.de>)
- The CORE ranking (<http://www.core.edu.au/conference-portal>)

The CORE ranking classify publication supports (conferences and journals) on several categories. In **coscinus** we restrict ourselves to publication supports in categories A*, A, B and C. Here is the meaning of the categories as summed up in <http://www.core.edu.au/conference-portal>:

- A* - flagship conference, a leading venue in a discipline area
- A - excellent conference, and highly respected in a discipline area
- B - good conference, and well regarded in a discipline area
- C - other ranked conference venues that meet minimum standards

For replicability, the data used for the experimentations described in this paper are available here: <http://coscinus.org/data>. The version of the DBLP database used for experimentations described in this paper is the 22 may 2018 one.

We use the last versions of the CORE ranking, i.e. CORE 2018 for conferences and CORE 2010 for journals. We provide also the CSV file linking the publication supports entries from DBLP and the ones from the CORE ranking that we use for coscinus and for the reported experiments. Since there is no standardized key naming of publication supports (globally based on acronyms of the support, but not always and with possible variations), that some conference names changed during their history, and that DBLP put some conferences under a same key, some publication supports are missing because of a mismatch of entry keys between DBLP and CORE.

For the data used in the coscinus project and for this paper, we have:

- 1 941 999 papers (CORE A*: 339 691 / A: 583 255 / B: 647 668 / C: 371 385)
- 1 172 474 authors
- 925 conferences (CORE A*: 63 / A: 201 / B: 308 / C: 353)
- 480 journals (CORE A*: 51 / A: 112 / B: 137 / C: 180)

Now let us turn to the way we exploit these data. The fundamental idea is that researchers in a field tend to publish in the same conferences/journals. And most of these researchers have at most one or two fields of competence. Hence, it is possible to draw some connections between conferences/journals by looking at how much researchers have papers in the two conferences/journals.

So we build a matrix of number of publications of each author in each support (conference/journals). Then, we build a matrix [supports \times supports], where we count the number of authors that publish in the two publication supports. To avoid too much noise we use a threshold to decide which authors are counted. It may happen that an author published once in a publication support that is not in his field. While it may be significant, there are also some chances that this publication is an “anomaly”, that is not related at all with the mainstream work of this author. This phenomenon induces some noise in the data. As an example, suppose that an author has 9 papers in conference A, 7 in conference B, 4 in conference C, and 1 in conference D. While we can suppose that conferences A, B and C are close, this is a much weaker conclusion than D is related to these three conferences. The MC (Minimal Count) threshold n works like this: we add 1 in the matrix position [supportA, supportB] for each author that has published at least n papers in supportA and n papers in supportB. Choosing high values for n allows to focus on mosts prolific authors only, and reduces the noise produced by anomalies.

The obtained counting matrix is the basic information we use as closeness measure between publication supports. There is just still one issue: some publication supports publish much more papers than others, then this matrix is biased by these differences in the total numbers of accepted papers. In order to fix this issue, we use a normalization step. This turns the first, counting matrix, in a second, proportional matrix. After this normalization step is applied, we obtain as final information the proportion of authors of a publication support that also published in another one.

There are three possible standard normalizations, since to normalize the data [support1, support2], one can normal-

ize on the total number of papers of support1 ($nsupport1 = [\text{support1}, \text{support1}]$), or the total number of papers of support2 ($nsupport2 = [\text{support2}, \text{support2}]$):

- min:[support1, support2] / min(nsupport1, nsupport2)
- max:[support1, support2] / max(nsupport1, nsupport2)
- avg:(min + max) / 2

Let us illustrate this process on a small part of the matrix with some AI conferences. The first matrix, the counting matrix is given on Table 1. This matrix is obtained from the real data from coscinus, using $MC = 3$, i.e. each unity in a matrix cell [supportA, supportB] means that an author published at least 3 papers in supportA and supportB. So one can see that there are 1826 authors that published at least 3 papers at IJCAI. And among these ones, 935 also published at least 3 papers at AAAI, whereas only 54 published at least 3 papers at JELIA.

Then we give the corresponding min-normalized matrix on Table 2. This is one of these three matrices (that we will call nmatrix), depending of the chosen normalization, that is used as input of both the principal component analysis and the clustering methods.

We perform a PCA (Principal Component Analysis) on the nmatrix matrix. PCA is a classical statistical method used to reduce the number of dimensions of a set of data [Pearson, 1901; Jackson, 2003; Jolliffe, 2002]. We consider our matrix as indicating a profile of each publication support on a space composed of 1405 dimensions (the total number of publication supports), and we project these publication supports on the 2 dimensions space chosen by the PCA method (the aim of the PCA is to find the best choice for a reduced number of dimensions).

This result is computed from the nmatrix for different values of MC (MinCount threshold), the three normalization functions (min, max, avg), and for different levels of publication supports according to CORE ranking classification (A*, A*+A, A*+A+B, A*+A+B+C). It is also possible, for example, to choose to display only A* or A*+A or A*+A+B publication supports when choosing the nmatrix based on A*+A+B+C publication supports.

We also perform clustering on the nmatrix matrix in order to identify clusters of publication supports. We propose two clustering methods :

- The first one is the k-means clustering method, that is one of the main clustering methods [Macqueen, 1967] (but it is not hierarchical, so clusters can be rather different when we increase the number of clusters).
- The second one is an agglomerative clustering method, implying that when we increase the number of cluster by one, one cluster will be splitted in two. More exactly we use the average-link clustering method [Ward, 1963; Rokach and Maimon, 2005].

The coscinus project allows to compute the results for conferences only, for journals only, and for both. From now on, in this study we will focus on conferences. We will also use the min-normalization in the following, and the k-means clustering method.

	LPNMR	ECML	JELIA	CP	AAAI	TARK	KR	NIPS	ICML	CADE	UAI	ECAI	AAMAS	IJCAI	AIML
LPNMR	141	2	46	5	43	0	34	2	2	3	3	34	17	55	0
ECML	2	516	3	5	94	0	2	86	125	4	26	44	27	93	2
JELIA	46	3	109	2	32	2	42	2	3	13	2	50	23	54	7
CP	5	5	2	361	126	3	20	13	9	15	17	70	34	122	1
AAAI	43	94	32	126	2496	15	128	271	281	28	163	203	390	935	4
TARK	0	0	2	3	15	53	11	1	0	2	6	6	13	18	5
KR	34	2	42	20	128	11	212	10	7	23	29	94	43	161	6
NIPS	2	86	2	13	271	1	10	1844	620	5	173	24	51	174	2
ICML	2	125	3	9	281	0	7	620	1088	4	143	30	50	206	2
CADE	3	4	13	15	28	2	23	5	4	343	2	28	8	56	8
UAI	3	26	2	17	163	6	29	173	143	2	460	33	43	130	2
ECAI	34	44	50	70	203	6	94	24	30	28	33	627	171	318	8
AAMAS	17	27	23	34	390	13	43	51	50	8	43	171	1421	317	7
IJCAI	55	93	54	122	935	18	161	174	206	56	130	318	317	1826	12
AIML	0	2	7	1	4	5	6	2	2	8	2	8	7	12	46

Table 1: Counting Matrix

	LPNMR	ECML	JELIA	CP	AAAI	TARK	KR	NIPS	ICML	CADE	UAI	ECAI	AAMAS	IJCAI	AIML
LPNMR	100	1.41	42.20	3.54	30.49	0	24.11	1.41	1.41	2.12	2.12	24.11	12.05	39.00	0
ECML	1.41	100	2.75	1.38	18.21	0	0.94	16.66	24.22	1.16	5.65	8.52	5.23	18.02	4.34
JELIA	42.20	2.75	100	1.83	29.35	3.77	38.53	1.83	2.75	11.92	1.83	45.87	21.10	49.54	15.21
CP	3.54	1.38	1.83	100	34.90	5.66	9.43	3.60	2.49	4.37	4.70	19.39	9.41	33.79	2.17
AAAI	30.49	18.21	29.35	34.90	100	28.30	60.37	14.69	25.82	8.16	35.43	32.37	27.44	51.20	8.69
TARK	0	0	3.77	5.66	28.30	100	20.75	1.88	0	3.77	11.32	11.32	24.52	33.96	10.86
KR	24.11	0.94	38.53	9.43	60.37	20.75	100	4.71	3.30	10.84	13.67	44.33	20.28	75.94	13.04
NIPS	1.41	16.66	1.83	3.60	14.69	1.88	4.71	100	56.98	1.45	37.60	3.82	3.58	9.52	4.34
ICML	1.41	24.22	2.75	2.49	25.82	0	3.30	56.98	100	1.16	31.08	4.78	4.59	18.93	4.34
CADE	2.12	1.16	11.92	4.37	8.16	3.77	10.84	1.45	1.16	100	0.58	8.16	2.33	16.32	17.39
UAI	2.12	5.65	1.83	4.70	35.43	11.32	13.67	37.60	31.08	0.58	100	7.17	9.34	28.26	4.34
ECAI	24.11	8.52	45.87	19.39	32.37	11.32	44.33	3.82	4.78	8.16	7.17	100	27.27	50.71	17.39
AAMAS	12.05	5.23	21.10	9.41	27.44	24.52	20.28	3.58	4.59	2.33	9.34	27.27	100	22.30	15.21
IJCAI	39.00	18.02	49.54	33.79	51.20	33.96	75.94	9.52	18.93	16.32	28.26	50.71	22.30	100	26.08
AIML	0	4.34	15.21	2.17	8.69	10.86	13.04	4.34	4.34	17.39	4.34	17.39	15.21	26.08	100

Table 2: Min-normalized Matrix

Let us illustrate what kind of information and vizualisation we can obtain from the general **coscinus** website, before focusing on AI conferences and on the historical study.

Figure 1 shows the 2-dimensional representation obtained by PCA (with MC=3, normalization=min), and the clusters obtained by k-means (for 12 clusters) of the A* conferences in the system (63 conferences).

One can note that the IJCAI cluster is composed of IJCAI, ICAPS, AAAI, AAMAS and KR. And that there is a “machine learning cluster” composed of NIPS, ICML, COLT,

UAI and RSS.

One can note also that there is a clear “Artificial Intelligence” region (these two clusters), and the other distinct region is the “Database + Web” one, that is clearly outside the main region with all other A* conferences.

These results are obtained by taking only the 63 A* conferences into account. We can obtain more detailed results by considering additionally A conferences. Figure 2 shows (under the same parameters) the A+A* conferences in the system (264 conferences). Figure 3 is a zoom on the IJCAI area, that

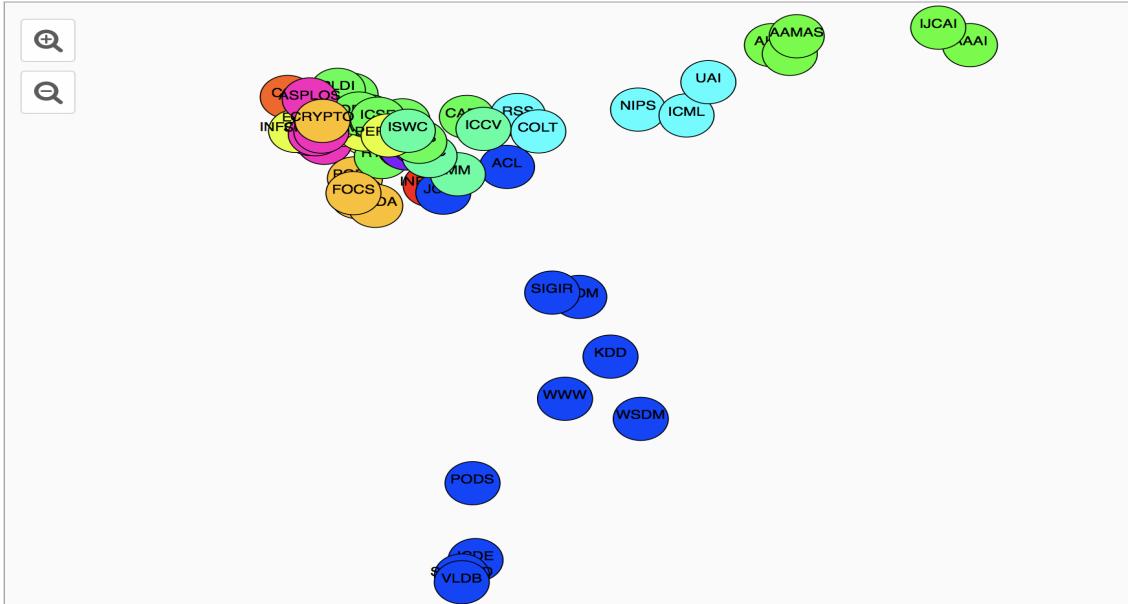


Figure 1: A^* conferences (MC=3, norm=min, 12 clusters)

allows to have a better visualization of the AI related part.

In this case the IJCAI cluster (still for 12 clusters for the k-means method) is composed of IJCAI, LPNMR, PKDD, JELIA, AISTATS, ICAPS, ICDM, AAAI, KDD, KR, NIPS, ICML, UAI, ECAI, CIKM, KCAP, WSDM, WWW, PAKDD, IDA, SIGIR, SDM, CVPR. So we obtain an unsurprising result, with, besides purely AI conferences, conferences on data-mining, web, and vision.

These results are obtained through the whole set of DBLP data. One interesting question is to study the historical evolution of conference closeness. This is what we address in the following (in order to have a better visualization and more information, the reader can go to <http://www.coscinus.org/ai>).

3 IJCAI Closest A^* Conferences

We will use a temporal window selection in order to study the historical evolution of closeness between conferences. We have chosen a 5 years window in order to have at least 2 editions of biennial conferences. We have centered the windows on the central year. So in the following when we will refer for instance to 2015, this means the 2013-2017 window period.

With a small temporal selection we can not use a big value for MC (Minimal Count). In the standard case (without temporal selection) we use MC=3 as default. Here, with 5-year windows, we select MC=1 in order to obtain sufficiently many data. One can consider that on a small temporal window the “anomalies” (i.e. a researcher publishing a paper in a conference outside her expertise domain) will not be numerous, so that the introduced noise will be small.

Let us illustrate in Table 3 the closeness between IJCAI and other A^* conferences, by looking at the conferences that are in the same cluster as IJCAI for 5, 10, 20 and 30 clusters.

For 20 and 30 clusters the conferences that appear the more often are AAAI and KR, and the ones that appear more than

once are AAAI, KR, ICAPS, AAMAS and UAI, so they can be considered as the closest conferences from IJCAI. But one can not really see a temporal evolution on this set (still it is possible to notice that in 2015 LICS and ISSAC appear with 20 clusters).

For 10 clusters, one can see mostly AI-related conferences. This means that there is clear boundary of the contours of AI. There are often discussions about what is AI, with a common sentence saying that “AI is everywhere”, i.e. that AI does not exist per se, and that it is closely intricated with other communities. But this shows that, as a publishing community, there is a clear contour of AI. This singularity can also be seen simply on the PCA result, since the AI conferences are grouped in one of the branches of the obtained representation.

For 5 and 10 clusters we can see conferences from other communities. In 1980 this is quite diverse, since appear a conference on information systems (ICIS), one of theoretical computer science (ISSAC) and one of cryptography (EUROCRYPT). In 1985 the only non-AI conferences are theoretical computer science conferences (LICS, POPL) and a distributed computing one (PODC). This can be maybe related to the developments of multi-agent systems, and of object/actor languages. In 1990 appear a verification conference (CAV) and an information retrieval one (SIGIR). In 1995 there are visualisation/vision/interface conferences (INFOVIS, ICCV, CHI, ISWC). Then from 2000 one can observe the convergence towards data mining and web conferences (KDD, SIGIR, WWW, WSDM), that lasts until now.

4 IJCAI-related AI Conferences

In this section we focus on a selection of AI conferences. We have chosen a set of A^* and A conferences from the CORE ranking. The 33 conferences we focus on are the following ones: AIML, FUZZIEEE, NIPS, ICAPS, PKDD, AAAI, IJ-

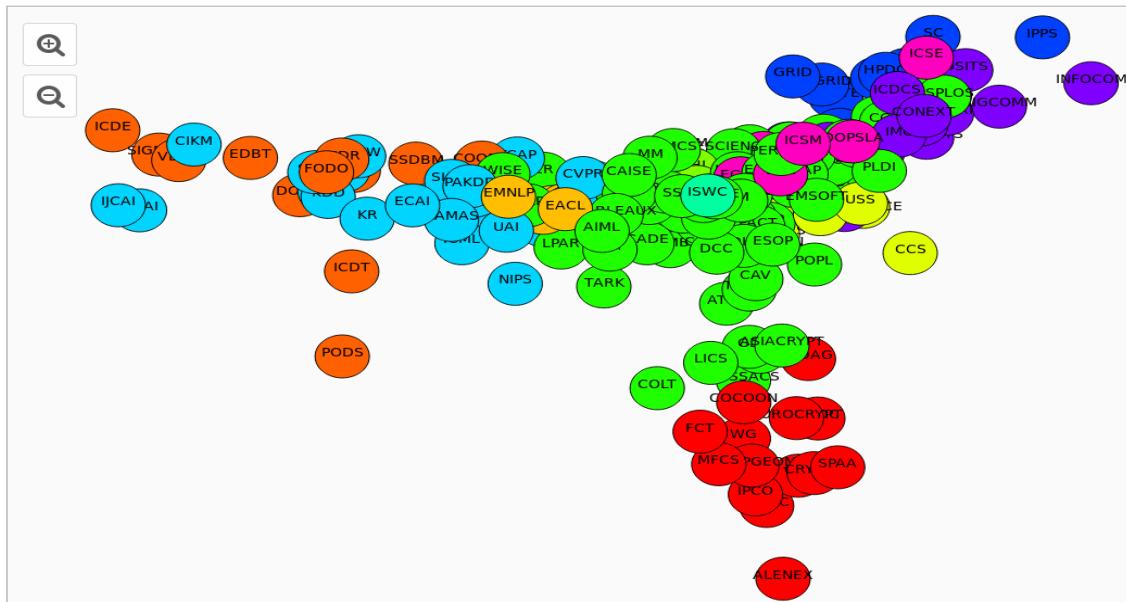


Figure 2: A+A* conferences (MC=3, norm=min, 12 clusters)

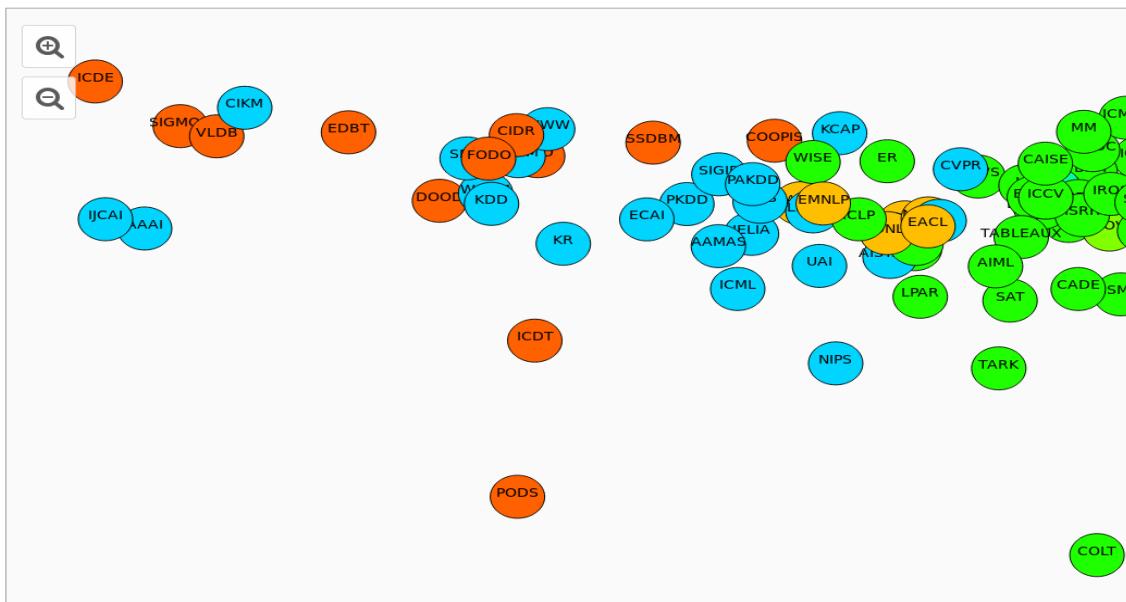


Figure 3: A+A* conferences - IJCAI area

CAI, UAI, ICONIP, ECAI, EMNLP, JELIA, TARK, ACL, TABLEAUX, CIKM, AISTATS, ICLP, ICDM, RSS, ICML, LPAR, ISRR, CP, KR, AAMAS, ESWS, IJCNN, CADE, COLT, SAT, LPNMR, FOGA.

The idea here is to try to identify AI sub-communities and their historical evolution. So in Table 4 we put all the clusters of these 33 conferences, when using the k-means clustering methods with 20 clusters and 30 clusters. This cluster computation is made on all computer science A* and A conferences (that represent 239 conferences), and then we only focus on

these 33 conferences and their clusters. This explains why we have less than 20 or 30 clusters in the 20/30 clusters column, since some clusters do not contain any of these 33 conferences.

A global comment on these results is that they illustrate the real singularity of artificial intelligence in the computer science community, since the clusters containing main AI conferences are mostly composed of AI conferences.

Let us first try to find conferences that are most of the times in a same cluster, that would suggest stable sub-communities.

year (± 2)	5 clusters	10 clusters	20 clusters	30 clusters
2015	ACL WSDM ICAPS ICDM AAAI WWW KDD SIGIR KR JCDL MM UAI ICCV AAMAS	ICAPS AAAI KR AAMAS	ISSAC KR CADE LICS	ICAPS AAAI KR AAMAS
2010	ACL WSDM ICAPS ICDM AAAI WWW KDD SIGIR KR NIPS ICML COLT UAI AAMAS	ACL WSDM ICAPS ICDM AAAI WWW KDD SIGIR KR FOGA NIPS ICML COLT UAI AAMAS	ICAPS AAAI	AAAI KR
2005	ICAPS AAAI KDD KR NIPS ICML COLT UAI	RSS ICAPS AAAI KR NIPS ICML COLT UAI ICCV AAMAS	AAAI KR	ICAPS UAI
2000	ICAPS AAAI KDD SIGIR KR NIPS ICML COLT UAI	RTSS CAV ICDM AAAI WWW KR ICML CADE MM UAI AAMAS	ICAPS AAAI UAI AAMAS	
1995	ACL KR CHI INFOVIS CADE ICCV AAMAS ISWC	ICAPS AAAI KDD KR FOGA NIPS ICML COLT UAI	ICAPS AAAI KR AAMAS	KR
1990	ACL CAV ICAPS AAAI SIGIR KR ICML CADE COLT UAI	ACL ICML CADE UAI	AAAI KR ICML	AAAI KR
1985	ACL CADE LICS	ACL PODC POPL AAAI STOC FOCS CADE LICS		
1980	AAAI ISSAC ICIS CADE EURO- CRYPT	ACL	AAAI	

Table 3: A* conferences in the same cluster than IJCAI (k-means clustering)

We can identify a first big group composed of AAAI, IJCAI, ECAI, KR and AAMAS. Then one can also identify a machine learning group (NIPS, COLT, ICML, AISTATS, UAI), a natural language one (ACL, EMNLP), an automated reasoning one (LPAR, CADE, TABLEAUX, LPNMR, SAT), and lastly a data-mining one (PKDD, CIKM, ICDM), but that seems less strongly connected than the other ones.

One can note that the conferences on neural networks IJCNN and ICONIP are not closely connected to NIPS.

One can note also that the robotics conference group (RSS, ISRR) is often connected to machine learning and neural networks conferences, but neither to the planning and scheduling one (AIPS).

It is also interesting to note that on the 30 clusters column, ICLP, CP and JELIA appear sometimes with the IJCAI group (AAAI, IJCAI, ECAI, KR, AAMAS), and sometimes in the automated reasoning group (LPAR, CADE, TABLEAUX, LPNMR, SAT), so this can be interpreted as being at the articulation between these two sub-communities (and this articulation is clear on the 20 clusters columns, where these two groups are usually unioned in a single one). It could be interesting to look at if some scientific thematic changes could explain the transition to one group to the other.

We do not have space to put here the corresponding landscapes obtained with PCA, but the reader can go on <http://www.coscinus.org/ai> to explore them.

5 Conclusion

We described the coscinus project, that allows to exploit DBLP data in order to compute proximity between publication supports, and that allows to study the contours of computer science communities. For this paper we added a temporal window selection feature and the possibility to focus on AI conferences only. We think that it can prove useful in order to study the evolution of computer science in general, and of artificial intelligence in particular. We made a very basic

use of this tool in this paper, but we believe that it could prove much more useful if combined with other observations.

Acknowledgments

This work has been partly supported by the project CPER DATA from the “Hauts-de-France” Region.

References

- [Jackson, 2003] J. Edward Jackson. *A User’s Guide to Principal Components*. Wiley Series in Probability and Statistics. Wiley-Interscience, 2003.
- [Jolliffe, 2002] Ian Jolliffe. *Principal component analysis*. Springer Verlag, 2002.
- [Macqueen, 1967] James Macqueen. Some methods for classification and analysis of multivariate observations. In *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.
- [Pearson, 1901] Karl Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.
- [Rokach and Maimon, 2005] Lior Rokach and Oded Maimon. Clustering methods. In Oded Maimon and Lior Rokach, editors, *The Data Mining and Knowledge Discovery Handbook*, pages 321–352. Springer, 2005.
- [Ward, 1963] Joe H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963.

year (± 2)	20 clusters	30 clusters
2015	ACL LPNMR JELIA AISTATS ICAPS AAAI TARK KR CP ICLP UAI ECAI EMNLP AAMAS IJCAI NIPS ICML COLT RSS ISRR CIKM ICDM FOGA AIML LPAR CADE SAT TABLEAUX PKDD EWS FUZZIEEEE IJCNN ICONIP	AISTATS NIPS ICML COLT UAI RSS ISRR CP LPAR CADE SAT TABLEAUX AIML FOGA LPNMR JELIA ICAPS AAAI TARK KR ICLP ECAI AAMAS IJCAI ACL PKDD ICDM CIKM EMNLP IJCNN EWS FUZZIEEEE
2010	LPNMR JELIA CP LPAR AAAI KR ICLP SAT TABLEAUX ECAI IJCAI AIML IJCNN ICONIP FOGA FUZZIEEEE CADE ICDM EWS ICAPS TARK AAMAS ACL EMNLP CIKM RSS PKDD NIPS ICML ISRR COLT UAI	FOGA PKDD ICDM NIPS ICML COLT UAI ACL EMNLP RSS ISRR CIKM SAT CADE ICAPS AAAI TARK ECAI AAMAS IJCAI LPNMR JELIA CP LPAR KR ICLP TABLEAUX IJCNN EWS FUZZIEEEE AIML
2005	PKDD AISTATS ICDM NIPS ICML COLT UAI LPNMR JELIA CP LPAR ICAPS AAAI TARK KR ICLP TABLEAUX ECAI AAMAS IJCAI AIML RSS IJCNN ICONIP ISRR SAT FUZZIEEEE FOGA EWS ACL EMNLP CIKM CADE	AISTATS NIPS ICML UAI CIKM FOGA ACL EMNLP COLT LPAR CADE SAT TABLEAUX PKDD ICDM RSS ISRR IJCNN EWS FUZZIEEEE ICONIP LPNMR JELIA CP ICAPS AAAI TARK KR ICLP ECAI AAMAS IJCAI AIML
2000	CIKM LPAR CADE ACL EMNLP ISRR LPNMR JELIA CP ICAPS AAAI KR ICLP TABLEAUX ECAI AAMAS IJCAI PKDD IJCNN AISTATS ICDM NIPS ICML COLT UAI ICONIP TARK FOGA FUZZIEEEE AIML	CIKM LPAR CADE TABLEAUX AIML FOGA ICAPS AAAI KR ECAI IJCAI CP TARK AAMAS LPNMR JELIA ICLP ACL EMNLP ICONIP ISRR FUZZIEEEE IJCNN PKDD AISTATS ICDM NIPS ICML COLT UAI
1995	LPNMR ILPS LPAR ICLP CADE TABLEAUX JELIA ICAPS AAAI KR ECAI AAMAS IJCAI AISTATS TARK ICML UAI ACL EMNLP PKDD CP FOGA NIPS COLT CIKM	LPNMR JELIA ILPS LPAR ICLP CADE TABLEAUX ICAPS AAAI TARK KR UAI ECAI AAMAS IJCAI CIKM COLT CP ACL EMNLP FOGA NIPS AISTATS ICML PKDD
1990	LPNMR JELIA ILPS LPAR KR ICLP CADE TABLEAUX ECAI IJCAI CIKM FOGA TARK ACL AAAI ICAPS IJCNN NIPS ICML COLT UAI	LPNMR JELIA AAAI TARK KR ECAI IJCAI CIKM IJCNN NIPS COLT FOGA ICML UAI ACL ILPS LPAR ICLP CADE TABLEAUX
1985	ACL AAAI UAI ILPS TARK ICLP CADE NIPS IJCAI ECAI	ECAI ILPS CADE UAI AAAI IJCAI ICLP NIPS TARK ACL
1980	AAAI IJCAI ECAI ACL CADE ICLP	AAAI IJCAI ECAI ACL CADE ICLP

Table 4: Clusters of A and A* AI conferences (k-means clustering)