Chapter 9

Different Versions of Hybrid Similarity Measures

The contextual similarity is defined by first taking the average similarity to the neighbors on either side, and then taking the average of these two. The combined similarity is defined as the average of the contextual and the content-based similarity. Nonetheless, there are other ways to combine all these similarities. Theoretical analysis discovers five different conceptual options to do so. Experimental research shows that there is no significant difference in results between any of them.

9.1 Introduction

When taking a closer look at the way in which the contextual similarity between elements v, w from a set of elements V is calculated (as described in Section 3.4) it can be seen that first one takes the average of all similarities from v to the neighbors of w, known as $\mathcal{S}_{neighbor}(v, w)$ from (3.2). Then, one does the same for all similarities from w to the neighbors of v, thus calculating $\mathcal{S}_{neighbor}(w, v)$. Finally, as described in (3.3), the average of these two values is taken to get the contextual similarity.

This means that $\mathcal{S}_{neighbor}(v, w)$ and $\mathcal{S}_{neighbor}(w, v)$ are contributing equally to $\mathcal{S}_{context}(v, w)$. This sounds like a fair deal. Since there is no demonstrable difference between v and w, why should $\mathcal{S}_{neighbor}(v, w)$ and $\mathcal{S}_{neighbor}(w, v)$ be treated differently? But what if the amount of neighbors that v has is much higher than the amount of neighbors that w has? In that case, the similarity between v and one of the few neighbors of w has a much greater contribution to the contextual similarity than the similarity between w and one of the many neighbors of v.

It can easily be seen that this situation may not be so desirable. As stated in Chapter 8, one of the benefits of the hybrid similarity measures is their ability to compensate for noisy information by taking the average of several similarities in the neighborhood defined by the relational information. The neighborhood similarity from one element to an element with a lot of neighbors is therefore much more reliable than the neighborhood similarity from that element to an element with only a few neighbors. This difference in reliability does not show in the calculation of the contextual similarity, since it regards the neighborhood similarity from one element to the other the same as the neighborhood similarity the other way around, regardless their reliability.

With this in mind, it may not be so fair at all that the contributions of $\mathcal{S}_{neighbor}(v, w)$ and $\mathcal{S}_{neighbor}(w, v)$ are equal. Much could be said for a hybrid similarity measure that is calculated in such a way that the contribution of a more reliable similarity is higher than that of a less reliable similarity. This can be solved easily by calculating a hybrid similarity between two elements as the average of all similarities from either element to the neighbors of the other. Then every similarity from one element to a neighbor of the other contributes equally, regardless of the amount of neighbors that element has.

This shows that there are more possibilities to create a hybrid similarity from the similarities between one element and the neighbors of the element between which the hybrid similarity is needed. It also shows that these new hybrids can have an interesting meaning, despite the fact that it is different from the original ideas. This raises the question of what different hybrid similarities can be created in comparable ways and which performs best.

In this chapter, we take a closer look at what the impact of those different choices is. Where this results in different similarities, we test if these similarities will lead to different results when applied in clustering or classification algorithms. From this, we will try to find the best way to calculate a similarity between two elements, based on their content and neighborhoods (if such best way exists).

The rest of this chapter is organized as follows. Section 9.2 analyses the contextual and combined similarities and from there, derives five meaningful and conceptually distinct manners to combine all used similarities into a hybrid similarity. Section 9.3 describes the experimental setup. Section 9.4 gives the results from these experiments, and Section 9.5 draws conclusions from these results.

9.2 More Similarities

First, the original hybrid similarities are analysed so that it will be easy to derive a variety of new hybrid similarity measures from it. Once again, consider the data set D defined as $D = (V, E, \alpha, \lambda)$ where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of n vertices or elements, $E \subseteq V \times V$ is the set of edges, $\alpha : V \to \mathcal{A}$ a function

that assigns to any element v of V an "annotation", and $\lambda : V \to \mathcal{L}$ a function that assigns to any element v of V a label. The annotation $\alpha(v)$ is considered to be the *content* of vertex v and the label $\lambda(v)$ is considered to be its *class*.

This data set format is the same as the one described in Section 3.3. Again, the space of possible annotations is left open and can be anything as long as it is possible to define a similarity measure $\mathcal{S}_{content} : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$ as a function that assigns a value to any pair of annotations expressing the similarity between these annotations. The space of possible labels is a set of m distinct elements, thus, $\mathcal{L} = \{l_1, l_2, \ldots, l_m\}$, each representing a specific class. It is possible that the class of a certain element is unknown.

Now, let $\mathcal{N}(w)$ be the set of all neighbors of w, so $u \in \mathcal{N}(w) \Leftrightarrow (w, u) \in E$, and let $\mathcal{T}(v, w)$ be the sum (or total) of all similarities from v to all neighbors of w, so $\mathcal{T}(v, w) = \sum_{u \in \mathcal{N}(w)} \mathcal{S}_{content}(v, u)$. To make things more clear, all definitions are also explained in Figure 9.1.

Using $\mathcal{N}$ and $\mathcal{T}$, the contextual similarity from (3.3) can be rewritten as:

$$\mathcal{S}_{contextual}(v, w) = \frac{\dfrac{\mathcal{T}(v, w)}{|\mathcal{N}(w)|} + \dfrac{\mathcal{T}(w, v)}{|\mathcal{N}(v)|}}{2} \tag{9.1}$$

Using this contextual similarity, the combined similarity from (3.4), with $c = \frac{1}{2}$, can be rewritten as:

$$\mathcal{S}_{combined}(v, w) = \frac{1}{2} \cdot \frac{\dfrac{\mathcal{T}(v, w)}{|\mathcal{N}(w)|} + \dfrac{\mathcal{T}(w, v)}{|\mathcal{N}(v)|}}{2} + \frac{1}{2} \cdot \mathcal{S}_{content}(v, w) \tag{9.2}$$

From here it is easy to see that:

$$\begin{aligned}
\mathcal{S}_{combined}(v, w) &= \frac{\dfrac{\mathcal{T}(v, w)}{|\mathcal{N}(w)|} + \dfrac{\mathcal{T}(w, v)}{|\mathcal{N}(v)|}}{4} + \frac{2 \cdot \mathcal{S}_{content}(v, w)}{4} \\
&= \frac{\dfrac{\mathcal{T}(v, w)}{|\mathcal{N}(w)|} + \mathcal{S}_{content}(v, w) + \dfrac{\mathcal{T}(w, v)}{|\mathcal{N}(v)|} + \mathcal{S}_{content}(w, v)}{4} \\
&= \frac{\dfrac{\dfrac{\mathcal{T}(v, w)}{|\mathcal{N}(w)|} + \mathcal{S}_{content}(v, w)}{2} + \dfrac{\dfrac{\mathcal{T}(w, v)}{|\mathcal{N}(v)|} + \mathcal{S}_{content}(w, v)}{2}}{2}
\end{aligned} \tag{9.3}$$

This shows that three different types of averages are calculated: first, the average of the similarities from one element to all neighbors of the other element is calculated. From now on, this will be known as the *neighborhood average*. Second, the average of the neighborhood average and the content-based similarity is calculated. When Figure 9.1 is divided into a right and a left half, it is

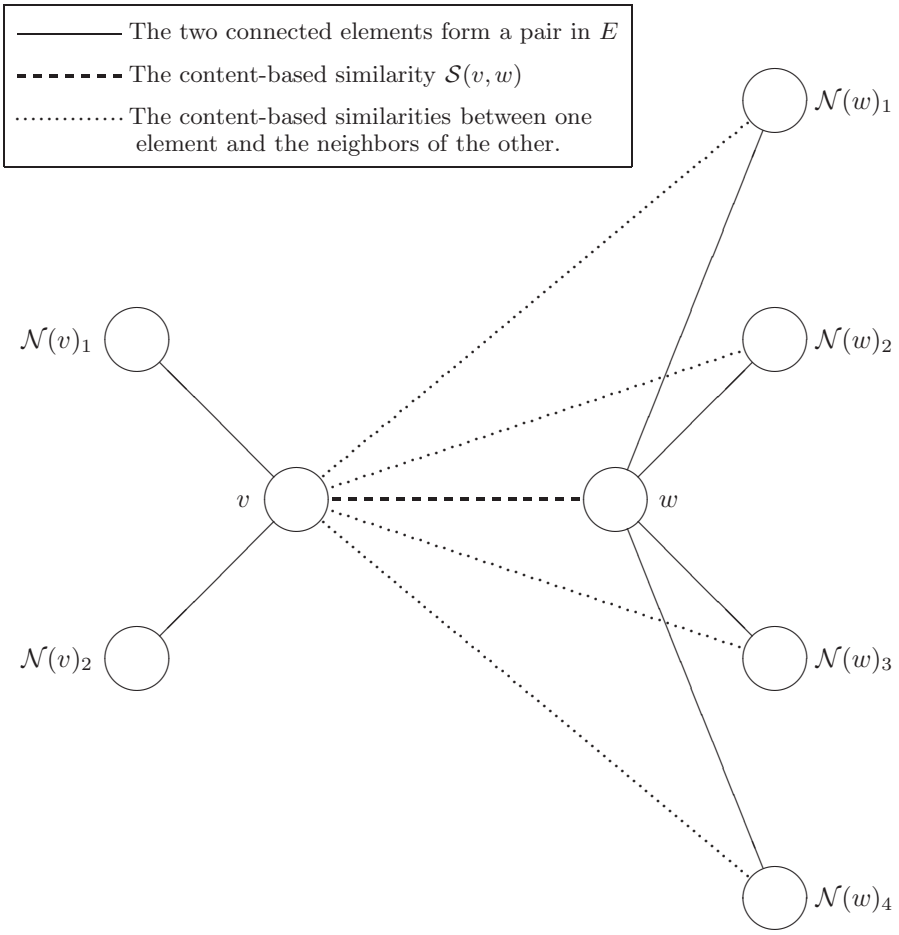

Figure 9.1: Overview of the objects necessary to calculate all different hybrid similarities between v and w. The solid lines represent the edges in the graph (i.e. the relational information from the dataset). Thus, an element labeled with $\mathcal{N}(x)_y$ is the y^{th} neighbor of x, with x being either v or w. The dashed line represents the content-based similarity between v and w ($\mathcal{S}_{content}(v, w)$). The dotted lines represent the content-based similarities between v and the neighbors of w ($\mathcal{N}(w)$). The sum of the similarities represented by dotted lines is $\mathcal{T}(v, w)$. Nota bene, the content-based similarities from w to all neighbors of v, needed to calculate $\mathcal{T}(w, v)$, are not drawn in this figure to keep it more clear. Also notice that it is possible that v and w are connected or that there is an element that is a neighbor of both v and w. Since this does not change the way to calculate any of the similarities, it is not taken into account here.

clear that this average is the average of all similarities from one element to all elements on the other side. Therefore from now on this will be known as the *side average*. Third, the average between the two sides is calculated. This will result in the average of the total amount of similarities and thus will be known as the *total average*.

It has been established that there are three places where an average can be taken: neighbor average, side average and total average. Also, it could be decided that on any of those places the average is not taken. In that case, one just sums the terms and takes the average in a later stadium. The only place where such a choice is not available is the total average. If this final average is not taken, the values for the resulting similarity could be out of range compared to the content-based similarity easily. To recall, the new hybrid similarity replaces the original content-based similarity in some data mining algorithm. If the algorithm uses similarities that are, for instance, between 0 and 1, then the values of the hybrid similarity also must be between 0 and 1.

This leaves two places where it could be decided whether or not to take the average at that point. Also, there is the choice of adding the content-based similarity between v and w, or only using the similarities between one of the nodes and the neighbors of the other. This leaves $2^{2+1} = 8$ possible hybrid similarities. Table 9.1 gives an overview of the resulting similarities from these combinations.

Table 9.1 clearly shows that, from the eight possible combinations, only five distinct hybrid similarities can be created. There are two reasons for this. First, when the content-based similarity between v and w is not included, the side average is calculated as an average only over the similarities from one element to the neighbors of the other, which is the same as the neighborhood average. In this case, there is no difference between taking the neighborhood average or taking the side average.

Second, if the neighborhood average is taken, then the neighborhood similarity (as calculated with (3.2)) could be considered as one similarity. If, in this case, the side average also is taken, then this will result in the neighborhood similarity being added to the content-based similarity and divided by 2. In the next step, the similarity for this side is added to the similarity of the other side and also divided by 2. If, on the other hand, the side average is not taken, then the two neighborhood averages are added to two times the content-based similarity. The resulting sum is divided by 4. The result of this is, of course, the same as when the side similarity was taken.

There are two different aspects for which the hybrid similarities are distinct. The first is whether or not the content-based similarity between v and w is inserted. This shows in the name of the hybrid similarity where a 'C' denotes the presence of the content-based similarity.

The second difference is how the average is calculated. It already has been shown that there is no difference between taking only the neighborhood average

#	$\mathcal{S}_{content}$	NEIGHBOR AVERAGE	SIDE AVERAGE	HYBRID SIMILARITY
1	no	no	no	$\mathcal{S}_{TA}$
2	no	no	yes	same as #3
3	no	yes	no	$\mathcal{S}_{NA}$
4	no	yes	yes	same as #3
5	yes	no	no	$\mathcal{S}_{CTA}$
6	yes	no	yes	$\mathcal{S}_{CSA}$
7	yes	yes	no	$\mathcal{S}_{CNA}$
8	yes	yes	yes	same as #7

Table 9.1: Different options for the hybrid similarities.

and taking both the neighborhood average and the side average. Also, it has been shown that the total average always needs to be taken. This leaves three possibilities: the one where the neighborhood average and the total average are taken, the one where the side average and the total average are taken, and the one where only the total average is taken. These three possibilities are distinguished by the first average that is taken and thus these averages show in the second part of the name of the hybrid similarities by, respectively: ('*NA*'), ('*SA*'), and ('*TA*').

The results in five different similarities, which we now discuss in detail.

No Content-Based Similarity and Neighborhood Average

For the first hybrid similarity, the content-based similarity between v and w is left out. Both the neighborhood average and the total average are taken, which results in the original contextual similarity as described in (9.1). Known as the Similarity without content-based similarity and Neighborhood Average ($\mathcal{S}_{NA}$), it is described as follows.

$$\mathcal{S}_{NA} = \frac{\dfrac{\mathcal{T}(v,w)}{|\mathcal{N}(w)|} + \dfrac{\mathcal{T}(w,v)}{|\mathcal{N}(v)|}}{2} \tag{9.4}$$

No Content-Based Similarity and Total Average

The second hybrid similarity is also without the content-based similarity between v and w, but here only the total average is taken. Known as the Similarity without content-based similarity and Total Average ($\mathcal{S}_{TA}$), it is described as follows.

$$\mathcal{S}_{TA} = \frac{\mathcal{T}(v,w) + \mathcal{T}(w,v)}{|\mathcal{N}(w)| + |\mathcal{N}(v)|} \tag{9.5}$$

With Content-Based Similarity and Neighborhood Average

The third hybrid similarity includes the content-based similarity between v and w. It takes both the neighborhood average and the total average, and it is the original combined similarity as described in (9.2). Known as the Similarity with Content-based similarity and Neighborhood Average ($\mathcal{S}_{CNA}$), it is described as follows.

$$\mathcal{S}_{CNA} = \frac{\frac{\mathcal{T}(v,w)}{|\mathcal{N}(w)|} + \mathcal{S}_{content}(v,w) + \frac{\mathcal{T}(w,v)}{|\mathcal{N}(v)|} + \mathcal{S}_{content}(w,v)}{4} \tag{9.6}$$

With Content-Based Similarity and Side Average

The fourth hybrid similarity uses the content-based similarity between v and w and both the side average and total average are taken. Known as the Similarity with Content-based similarity and Side Average ($\mathcal{S}_{CSA}$), it is described as follows.

$$\mathcal{S}_{CSA} = \frac{\frac{\mathcal{T}(v,w) + \mathcal{S}_{content}(v,w)}{|\mathcal{N}(w)| + 1} + \frac{\mathcal{T}(w,v) + \mathcal{S}_{content}(w,v)}{|\mathcal{N}(v)| + 1}}{2} \tag{9.7}$$

With Content-Based Similarity and Total Average

The fifth and final hybrid similarity also uses the content-based similarity between v and w, and only calculates the average at the very end where the grand total of all similarities is one big sum. Known as the Similarity with Content-based similarity and Total Average ($\mathcal{S}_{CTA}$), it is described as follows.

$$\mathcal{S}_{CTA} = \frac{\mathcal{T}(v,w) + \mathcal{S}_{content}(v,w) + \mathcal{T}(w,v) + \mathcal{S}_{content}(w,v)}{|\mathcal{N}(w)| + |\mathcal{N}(v)| + 2} \tag{9.8}$$

9.3 Experimental Setup

The five hybrid similarities need to be compared to the original content-based similarity as well as amongst each other. To do so, several experiments were conducted on the five subsets, CORA-1 through CORA-5 of the Cora data set [65] as described in Section 5.2. For three different data mining algorithms, the original content-based similarity was replaced by any of the five hybrid similarities and the effect on the performance was measured. Since all three algorithms were described more elaborately in previous chapters, only a short description is given here.

The first algorithm used is *agglomerative hierarchical clustering*, as previously described in Section 5.3 and in, for instance, the book by Tan et al. [94].

In short, agglomerative hierarchical clustering is a clustering algorithm that starts with every element in its own cluster. Then, the algorithm iteratively merges the two closest clusters until only one cluster remains. During these experimants, average linkage is used.

The second algorithm used is *k-medoids* from Kaufman and Rousseeuw [45]. It is previously described in Section 6.2.1. In short, k-medoids is a clustering algorithm that starts by selecting k random elements which start as the center, or medoid, of the clusters. Then, iteratively, every other element is placed in the cluster with the mean it is most similar to. Once all elements are assigned, the medoids are recalculated as the element in a cluster that is most similar to all others. The algorithm halts when nothing changes anymore.

The clusterings found in the above mentioned clustering algorithms are evaluated by calculating the F-score by Larsen and Aone [56]. For agglomerative hierarchical clustering this F-score is calculated every 10 steps. For k-medoids, k was varied from 100 to 2 and for every k the experiments were repeated 100 times to compensate for the effects caused by randomly choosing the initial medoids.

The third algorithm used is KNN-*classification*, as described in Section 7.2 and in, for instance, the book by Tan et al. [94]. In short, KNN-classification is a classification algorithm that tries to predict a label for a certain element. It does so by regarding the labels of the k most similar elements and choosing the label that is most common as a prediction. The ratio of correctly predicted elements defines how well it classifies.

9.4 Results

Figure 9.2 shows two examples of the results as found by the clustering algorithms: k-medoids for subset CORA-1 and agglomerative hierarchical clustering for subset CORA-3. The results of other subsets for these algorithms show similar characteristics.

The graph describing the results for k-medoids on CORA-1 shows that clustering with the original content-based similarity gives significantly weaker results than using any of the hybrid similarities. Furthermore, there is no significant difference in performance between any of the hybrid similarities.

The same can be said when regarding the results for agglomerative hierarchical clustering on CORA-3. At the beginning of the clustering algorithm, the F-scores do not differ much. The content-based similarity is even slightly better than the others. But, as the algorithm proceeds, and the F-scores rise, the content-based similarity is the first that reaches its maximum. The hybrid similarities reach their maximum at a later point, where it is also higher.

Figure 9.3 shows the results for KNN-classification on subset CORA-5. The first graph shows the results for k in the range from 1 to 7918 which is the size of CORA-5. The second graph shows the same results, but then magnified.

Here, k ranges from 1 to 40 on the horizontal axis and the vertical axis is adjusted accordingly. Once again, the difference in performance between the original similarity and the five hybrids is very obvious. Also here, there is no real significant difference in performance between the five hybrids.

Table 9.2 gives an overview of all the best results found. For every clustering method/dataset-combination, it is always the original, content-based similarity that performs the worst. The five hybrid similarities are always significantly better. The one exception might be $\mathcal{S}_{CTA}$ with k-medoids on CORA-5, which is only slightly better than $\mathcal{S}_{content}$ and closer to this content-based similarity than to the other hybrids. But, as it performs similar to the other hybrids on the other settings, it can be considered an exception.

9.5 Conclusions

The contextual similarity and the combined similarity from Chapter 3 were created with the use of the similarities to the neighbors of an element. We argued that there are five conceptual possibilities to combine all these similarities to a hybrid similarity. These five hybrid similarities have been used on three different clustering techniques: agglomerative hierarchical clustering, k-medoids, and KNN-classification. Experiments on subsets of the Cora dataset showed that:

- The hybrid similarities outperform the original, content-based similarity in every setting.
- There is no significant difference between the hybrid similarities.

From this, it can be concluded that it does lead to better results when a content-based similarity is replaced by a hybrid similarity (i.e. a similarity that is a combination of the similarities from one element to the neighbors of the other). However, it does not matter in which way these similarities are combined.

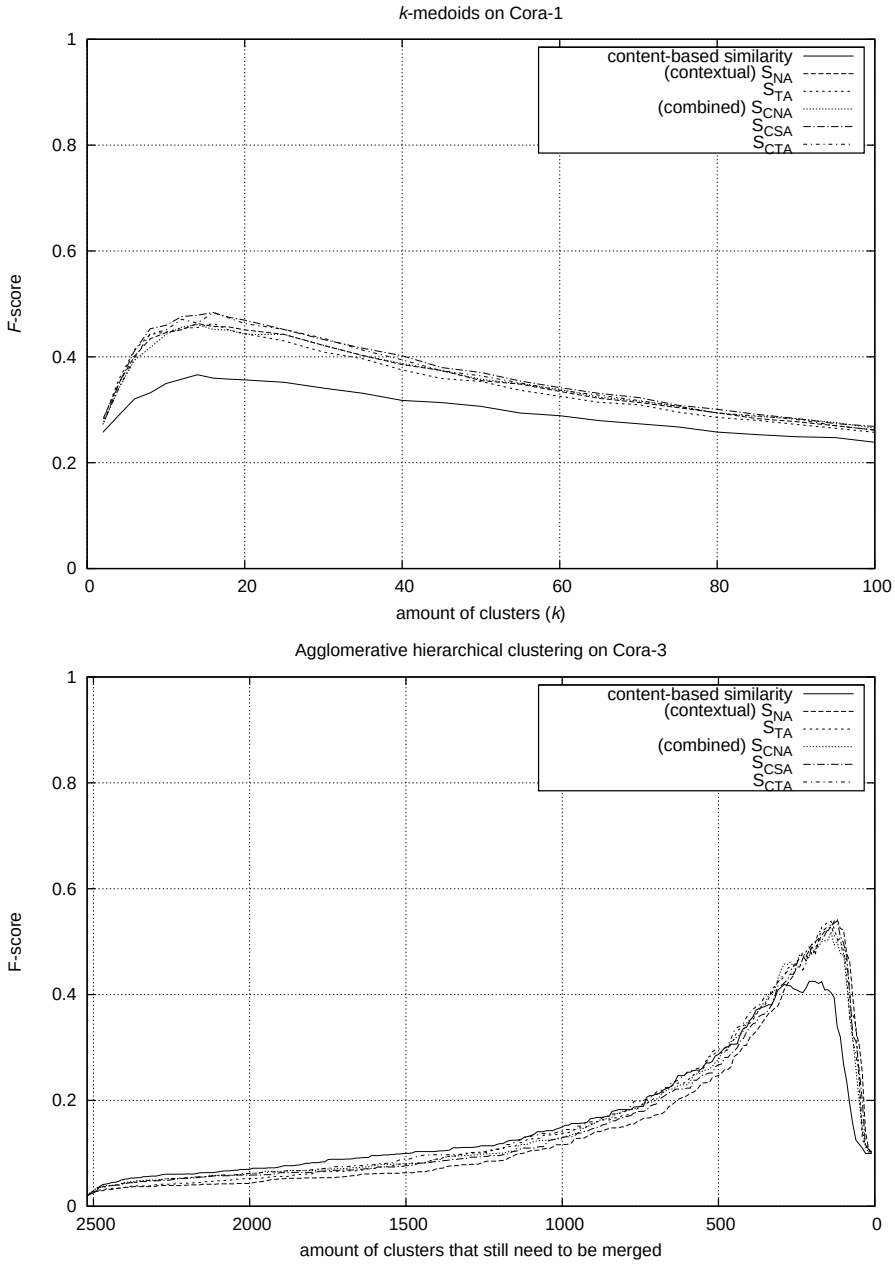

Figure 9.2: Clustering quality for k-medoids on CORA-1 (top) and agglomerative hierarchical clustering on CORA-3 (bottom) for the five hybrid similarities compared to the content-based similarity.

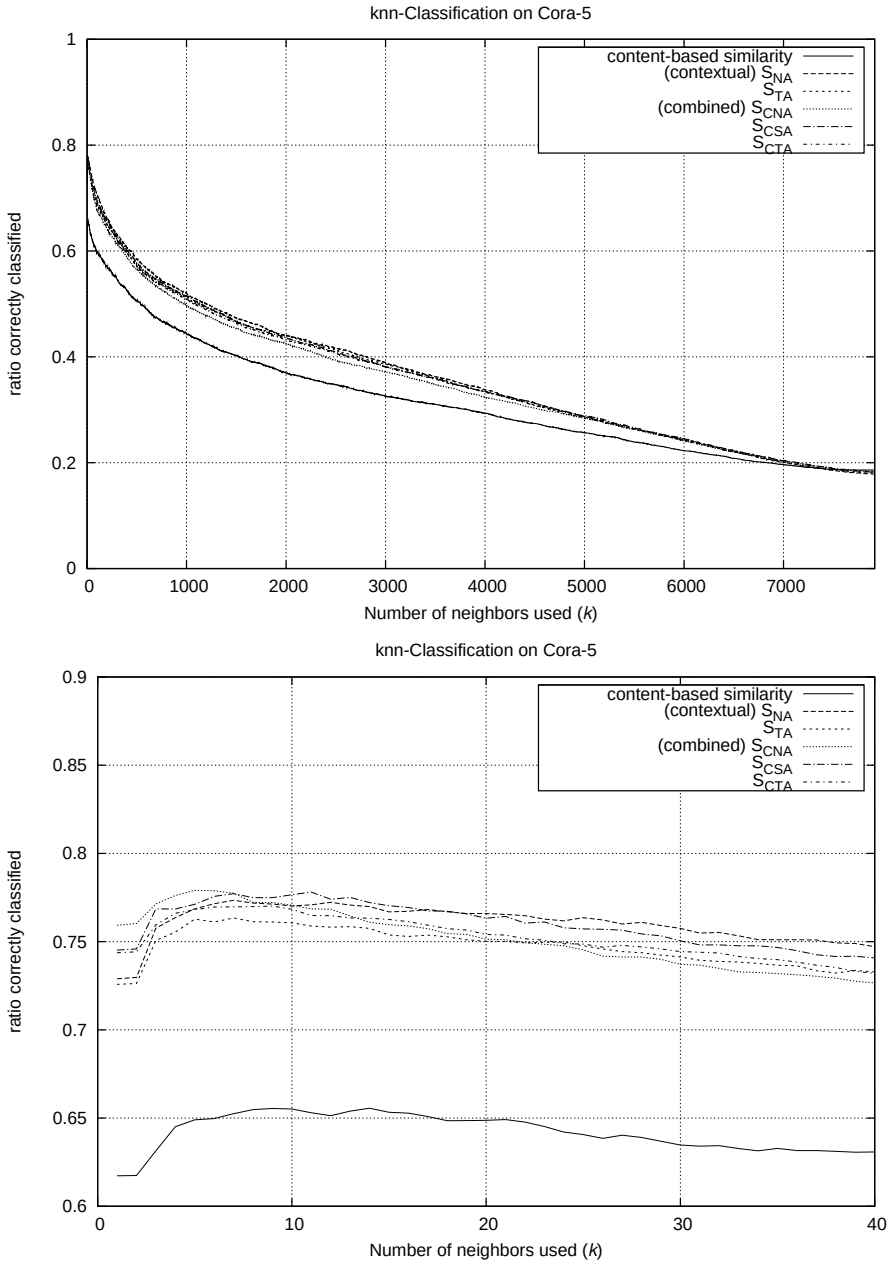

Figure 9.3: Ratio of correctly classified elements on CORA-5 with KNN-classification with k from 1 to 7916 (top) and zoomed in on k from 1 to 40 (bottom) for the five hybrid similarities compared to the original content-based similarity.

Agglomerative Hierarchical Clustering

Similarity	Cora-1	Cora-2	Cora-3	Cora-4	Cora-5
$\mathcal{S}_{content}$	0.58	0.44	0.42	0.44	0.38
$\mathcal{S}_{NA}$	0.63	0.58	0.54	0.50	0.45
$\mathcal{S}_{TA}$	0.63	0.58	0.54	0.50	0.46
$\mathcal{S}_{CNA}$	0.67	0.54	0.52	0.50	0.44
$\mathcal{S}_{CSA}$	0.64	0.58	0.54	0.49	0.46
$\mathcal{S}_{CTA}$	0.66	0.53	0.53	0.52	0.45

K-medoids

Similarity	Cora-1	Cora-2	Cora-3	Cora-4	Cora-5
$\mathcal{S}_{content}$	0.37	0.27	0.24	0.22	0.18
$\mathcal{S}_{NA}$	0.46	0.39	0.36	0.33	0.30
$\mathcal{S}_{TA}$	0.46	0.39	0.36	0.34	0.31
$\mathcal{S}_{CNA}$	0.46	0.37	0.34	0.32	0.28
$\mathcal{S}_{CSA}$	0.48	0.40	0.37	0.35	0.32
$\mathcal{S}_{CTA}$	0.48	0.40	0.38	0.35	0.22

knn-Classifiaction

Similarity	Cora-1	Cora-2	Cora-3	Cora-4	Cora-5
$\mathcal{S}_{content}$	0.83	0.75	0.68	0.69	0.66
$\mathcal{S}_{NA}$	0.93	0.88	0.82	0.82	0.77
$\mathcal{S}_{TA}$	0.92	0.87	0.82	0.80	0.76
$\mathcal{S}_{CNA}$	0.93	0.88	0.82	0.82	0.78
$\mathcal{S}_{CSA}$	0.93	0.88	0.82	0.81	0.78
$\mathcal{S}_{CTA}$	0.92	0.87	0.82	0.81	0.77

Table 9.2: Best results found for each clustering method, similarity, and subset. For agglomerative hierarchical clustering and k-medoids, this score is the best found F-score, and for knn-classification this score is the ratio of correctly predicted elements.