An Aggregation Procedure for Building Episodic Memory

Olivier Ferret
LIMSI-CNRS
BP 133
91 403 Orsay CEDEX
FRANCE

Brigitte Grau
LIMSI-CNRS and IIE-CNAM
BP 133
91 403 Orsay CEDEX
FRANCE

Abstract

When dealing with narrative texts, a system must possess a strong domain theory, and especially knowledge about situations occurring in the world. Otherwise the system must envisage comprehension as a complex process including learning from the texts themselves to improve its capabilities. This requires managing past solutions and completing them when analoguous situations happen in other texts in order to create general situations. We propose a system, MLK (Memorization for Learning Knowledge), that organizes specific situations in an episodic memory by aggregating the similar ones in a single unit. This aggregation process leads to a progressive enrichment and generalization of the overall situations and of their specific features. MLK is a system conceived to allow the emergence of structures, their accessing being realized by a propagation process. Therefore, with MLK, we are able to address the problem of understanding and learning even when a domain theory is lacking.

1 Introduction

Modelling and using the experience of a system is central for text understanding as the machine does not possess all the necessary background knowledge. So if a system can take advantage of the information present in texts and reuse it, it will be able to improve itself. Psychological studies [Vygotsky, 1962; Bartlett, 1932] show that, when experiences are memorized, they are grouped according to resemblance criteria to form a unit, and then abstract classes emerge progressively. The possibility of using known situations related to a new one for understanding it, allows a human to complete his own knowledge about this situation and to improve his capacities of understanding. A system applying such a method is permanently evolving and as a result, a same story may be analyzed in different ways at two different times. We are mostly interested in pragmatic knowledge about concrete situations occurring in the world, traditionally represented by schemata. This kind of knowledge is necessary to integrate sentences in a context and to infer chains of causal relations between events.

Many studies have been carried out on memorizing, understanding and learning as components of a whole process. These systems learn new specialized situations, SWALE [Schank and Leake, 1989] and AQUA [Ram, 1993] by explanation based learning (EBL) and OCCAM [Pazzani, 1988] by integrating empirical learning and EBL. A common point in these systems is the presence of general knowledge to guide the explanation process.

In contrast, our goal is to build a system able to work even when lacking general pragmatic knowledge about concrete situations. In this context, text analysis relies on linguistic clues, such as causal and temporal information, and on other known similar situations if they exist. These weak methods do not lead to a causal explanation of all the events of a text, and thus their pertinence is not fully justified. The presence of an event in two particular cases is not sufficient to conclude if it belongs to a possible general situation. Only the conjunction of its recurrence and the presence of causal clues in different texts will make its abstraction possible. Furthermore, all the events describing a situation are not specified in a text, but those which are relevant to the story. That is why a learning process must recognize similar situations to complete them text after text and not just memorize these situations independently of each other.

To allow the emergence of pertinent features from all similar situations, our system, MLK, uses incremental learning. When memorizing, MLK groups similar situations together to form an aggregated structure, in which similar features are reinforced while rare events become weaker. This aggregation process leads to a progressive enrichment and generalization of the overall situations and of their specific features. This kind of memorization is conceived to build structures on which abstraction of general units will be based and to provide an adaptative retrieval process by propagating activation. The retrieval of situations analoguous to a new one is always dependent on the state of the memory. This approach has several advantages related to the management and the retrieval of similar cases and provides a more psychologically-plausible model of memorization and learning when general background knowledge is lacking.

2 The MLK system

MLK is initially given a semantic representation of sentences organized in situations. The segmentation is currently done by hand. We use linguistic criteria (causal clues, order of sentences and temporal markers) plus the type of the predicates in each clause. Situations are named thematic

units (TUs) because they group the specific knowledge about a topic. Representations of texts form episodes: a set of TUs related by thematic links, such as topic shift (another subject) and topic deviation (a particular point). The episodes define more global contexts containing specific situations. A TU is a causal structuration of events. Features contained in each slot are represented by conceptual graphs [Sowa, 1984]. They are derived from the semantic knowledge of the system which is represented by a lattice of types of concepts associated to canonical and definition graphs. The canonical graphs contain semantic constraints on the casual roles fillers. In definition graphs, more precision about the roles is available, as well as about the predicate consequences.

The aggregation process is based on similarities between situations. Two situations are similar if their causal structure and their features are analoguous. A similarity measure has been defined on the basis of the structure itself and of its features. The process entails grouping similar features and reinforcing them by augmenting their weight. Other features are simply added. The reinforcement of some elements implies the weakening of others. By iterating this process, a situation will be progressively completed and the most recurrent features describing it will emerge. They will be the candidates for the description of the abstracted situation.

We consider short narrative texts, such as those found in newspapers, that may reference more than one situation. For example, a murder attempt, an aggression, a quarrel, political murders: they all describe analoguous situations that lead to progressively build a single aggregated TU about an attempted murder, described in figure 3 (section 3).

The aggregation process also accounts for the case retrieval problem. In most CBR systems, cases are indexed by predetermined features. In [Kolodner and Simpson, 1989], cases are classified according to general situations and the access mode is based on the traversal of a network of discriminating indices. In AQUA, cases are indexed by several features, especially by the general situations (MOPs) involved. But when lacking general situations, the system does not know what it is learning about and therefore no indices can be used. Another approach is to perform the retrieval of cases by a propagation process, as presented in REMIND [Lange and Wharton, 1993]. Propagation requires weighing the features of the structure. In REMIND, this task is made by hand when the domain knowledge is given. In MLK, the aggregation leads to computing all the weights and to dynamically modifying them according to the experience. Retrieval criteria evolve while the memory content changes without maintaining an explicit structure of indexation features. The problem of deciding what features are relevant for retrieving a structure does not occur. The spreading activation network encodes knowledge about concepts and situations. Concepts can be viewed as indices for situations, because they are part of the conceptual graphs that represent the features of the TUs. Furthermore they create links between TUs other than the thematic links.

When analyzing a text, sentences are decomposed in clauses and the MLK retrieval process is given each concept of a clause. Units in the network representing these concepts are clamped to high level of activation. Activation then spreads into the network. As the activation function encodes the prior context, situations that fit with the current clause and are coherent with the context are highly activated.

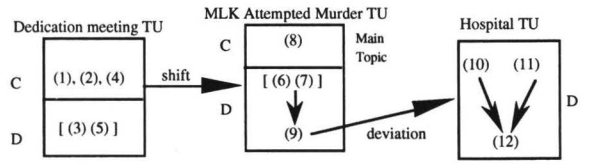
Thus the aggregation mechanism provides an answer to the organization and structuration of cases in an episodic memory for the purpose of learning from texts, even when general situations do not guide the process. Our model satisfies two constraints: it allows incremental learning independently of the domain and is a case basis for CBR.

3 The aggregating episodic memory

3.1 Representation of TUs and encoding of episodes

A text is represented by an episode which is a structured set of TUs. One of them encodes the main topic of the episode and the others are linked to this main situation by thematic links. TUs can be considered as instances of schemata. They encode causal and temporal knowledge between events of a specific situation issued from a text analysis. Their representation formalism is derived from the schema representation chosen, which is close to MOPs [Schank, 1982]. Figure 1 shows the segmentation of a text, the distribution of the clauses in the TUs and details one TU.

A few years ago, **[**I was in a department store in Harlem**](1)** **[**with a few hundred people around me**](2)**. **[**I was signing copies of my book "Stride toward Freedom"**](3)** **[**which relates the boycott of buses in Montgomery in 1955-56**](4)**. Suddenly, while **[**I was appending ma signature to a page**](5)**, **[**I felt a pointed thing sinking brutally into my chest**](6)**. **[**I had just been stabbed with a paper knife by a woman**](7)** **[**who was acknowledged as mad afterwards**](8)**. **[**I was taken immediately to the Harlem Hospital**](9)** **[**where I stayed on a bed during long hours**](10)** while **[**many preparations were made**](11)** **[**in order to remove the weapon from my body**](12)**.



Circumstances	
[Located] —	***[Mad]*** —
(experiencer) —> [Man: MLK]	(experiencer) —> [Woman]
(location) —> [Store]	
Description	
[Stab] —	***[Take_Hospital]*** —
(agent) —> [Woman]	(agent) —> [HumanBeeing]
(recipient) —> [Chest] —	(patient) —> [Man: MLK]
(part) —> [Man: MLK]	(destination) —> [Hospital]
(instrument) —> [PaperKnife]	(manner) —> [Quickly]
(manner) —> [Brutally]	
Outcomes	
[Wounded] —	***[Patient]*** —
(experiencer) —> [Man: MLK]	(experiencer) —> [Man: MLK]

Figure 1: A text and its representation

A TU is described by three slots: Circumstances, Description and Outcomes. The Circumstances slot is valued by a set of conceptual graphs that represent the states known when the situation occurs. Conceptual graphs in the Description slot

represent the events which take place and are partially ordered by chronological links. The Outcomes slot contains the resulting states and is also valued by a set of graphs. We can see that some conceptual graphs are issued from semantic inferences, as Patient and Wounded. The conceptual graphs are build around a predicate. The predicate is the central concept when describing an event; its type is a subtype of action, process or state. The other concepts fill in the casual roles linked to the predicate. Their types must be specializations of the types coming from the canonical graphs (we call these latter types 'reference types').

3.2 Similarity

The aggregation process requires that MLK find known situations in memory similar to the new TUs. Situations in memory result from preceding aggregations and are named aggregated TUs. The aggregation takes place at two levels. At the episode level, its purpose is to make the global contexts emerge by the means of recurrent sets of situations. At the TU level, its purpose is to learn about a situation by incrementally completing it while recurrent events emerge. At both levels, the aggregation process has to evaluate the similarity of TUs with aggregated ones.

For this purpose, the similarity measure we use [Ferret and Grau, 1995] is decomposed in two steps. Global similarity is based on the number of similar events, according to the slots they belong to. The exact degrees of similarity of slots and events are examined when this global similarity gives a medium result. Two events are similar if their predicates contain the same type and if at least one of their concepts filling casual roles is similar. This occurs if their types share a supertype inferior to the reference type of the casual role. The degree of similarity of events and TUs is computed respectively on the basis of the weights of concepts and graphs. An example is given in section 5.

This similarity measure results in grouping features that belong to a same topic without consideration of a description level. Situations in texts which refer to the same topic are often different specializations in which some events are the same but roles are different. An aggregated TU is conceived to be the basic structure on which abstraction will rest. This latter process will be in charge to eventually split a memorized situation into several units, general and specialized schemata, according to the weights of the events and concepts. If the similarity criteria were more strict, each aggregated TU would be too specific and the common points of a general unit would not emerge.

3.3 Episode aggregation

At the episode level, MLK reinforces configurations of similar situations. Aggregation in these cases consists in the reinforcement of the common thematic links and the aggregation of the related TUs. At the TU level, the graphs of events that have the same predicate are aggregated; the others are just added. Same causal or temporal relations between graphs are also reinforced. Aggregation of graphs leads to progressively generalizing their concepts.

As previously mentioned, aggregated TUs result from successive aggregations in the episodic memory. So, they are structures like the TUs, whose features are weighed aggregated graphs. This kind of graphs is defined as an extension of conceptual graphs in such a way as to maintain the capability of applying the basic derivational operations. We have added to them the similarity computation and the aggregation process. The aggregation process is defined as a maximal join where predicates must join: same casual roles are joined and their types of concepts are abstracted as in figure 2. New relations and concepts are then added.

[*Stab*](1.0) —
- **(agent) (1.0) —>** **[Man] (1.0)**
  - (agent) [1,2] — Soldier [1], Young-Man [2]
- **(recipient) (1.0) —>** **[Body] (1.0) —**
  - (recipient) [1,2] — Arm [1], Stomach [2]
  - **(part) (1.0) —>** **[Man] (1.0)**
    - (part) [1,2] — Head-of-State [1], Young-Man [2]
- **(instrument) (1.0) —>** **[Knife] (1.0)**
  - (instrument) [1,2] — Bayonet [1], Flick knife [2]

Figure 2: An aggregated graph

Types of aggregated concepts are computed at each aggregation. They are the most specific concept abstraction (msca), different from the reference type, of all the aggregated instances. If such a type does not exist in the lattice, we choose the msca that groups the maximum of instances if it is possible. To compute it, each aggregated concept keeps the instances it comes from. For each of these instances, we also keep the specific episode it is part of (as [1] in the figure). Thus, we can rebuild all the specific situations which have contributed to the formation of an aggregated TU. This capability is preserved to enable the retrieving of specific links between graphs. It will be useful to justify a reasoning based on the aggregated situation, TU aggregation being done regardless of shared instances between graphs. For example, if an agent of an event is also the patient of another event in the same situation, this piece of knowledge is not maintained at the aggregation level since each aggregated concept type evolves inside its own graph regardless of the evolution in the other graphs. This kind of knowledge is also basic to schemata abstraction; it prepares the formation of roles. The last characteristic added on the aggregated concepts is a counter to compute their weight. Weights on the predicates represent the importance of the event inside the TU and weights on the other concepts, the importance of the concept in the graph. These counters are equal to the number of aggregations. Thus, the weight of an event is computed by dividing the counter of the predicate by the number of times the TU has been aggregated. Similarly, the weight of a concept is the counter of the concept divided by the number of aggregations of the predicate. Weights on relations between two graphs are the number of aggregations divided by the number of common episodes between the two graphs.

The following example, in which five TUs have been aggregated, only shows the predicates of the graphs with their weights and the episodes they belong to. For presentation purpose, chronological relations between events in the

Circumstances (C)	
(a) Located **(0.2)** [1]	*(b) Disagree* **(0.4)** [2,3]
(c) Live **(0.4)** [3,5]	*(d) Have_Bath* **(0.2)** [3]
(e) Threatened **(0.2)** [3]	*(f) Commanding* **(0.2)** [4]
(g) Sleeping **(0.2)** [5]	
Description (D)	
(a) Attack **(0.4)** [1,5]	*(b) Stumble* **(0.2)** [1]
(c) Stab **(1.0)** [1,2,3,4,5]	*(d) Arrest* **(0.4)** [1,2]
(e) Hit **(0.2)** [2]	*(f) Enter* **(0.4)** [3,5]
(g) Lose **(0.2)** [4]	*(h) Attach* **(0.2)** [5]
(i)Tear **(0.2)** [5]	
Outcomes (O)	
(a) Imprisoned **(0.5)** [1,2,5]	*(b) Wounded* **(0.4)** [1,5]
(c) Dead **(0.5)** [2,3,4]	*(d) Guillotined* **(0.2)** [3]

Inter-graphs relations: D.d -> O.a **(1.0)** [1, 2]
D.c -> O.b **(1.0)** [1,5], O.c **(1.0)** [2,3,4]

Figure 3: The aggregated TU Attempted_Murder

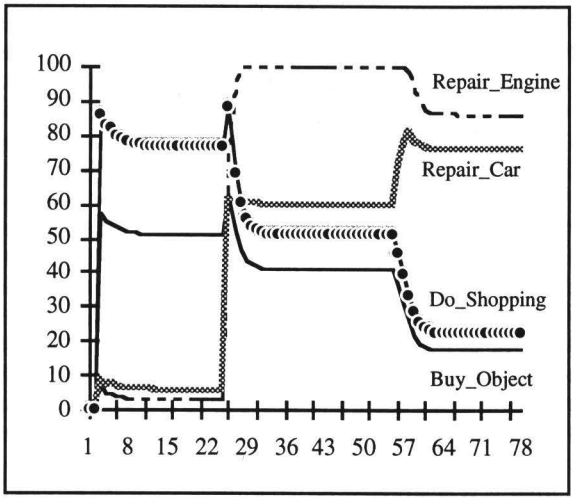


Figure 4: Activation level of aggregated graphs

description slot and the aggregated concepts have not been mentioned. Only causal relations between actions and their results have been precised. States setting up a situation are extremely various in texts, and often without explicit causal relations; so they are all considered as possible circumstances. At this state of the memorization, none emerge completely, but we can note that some are semantically close: *Located* vs *Live* and *Disagree* vs *Threatened.* Events in the description slot are more interesting with a characteristic event always reinforced, *Stab*. Others have to be confirmed before deciding their relevance, like *Arrest.* Notice that the reinforcement of a causal relation between two events is also an indication of their interest. Weak weights on events can be interpreted as anecdotes (*Stumble*) or events that do not really belong to the situation. Outcomes are often linked to an action and are significant. A discussion about the formation of this TU is presented in section 5.

4 The retrieval process

A spreading activation mechanism makes possible the use of episodic memory as an associative memory in order to select the relevant knowledge for processing the clauses coming from the texts. The figure 4 shows activation of four aggregated graphs resulting from the network after it successively receives as input the clauses of the following little story:

Yesterday, I went out to do shopping. I bought a spare part to repair my car. But finally, I let the mechanic repair it.

The two aggregated graphs, BuyObject and Shopping, are part of an aggregated TU called *GoingSupermarket.* RepairCar and RepairEngine are part of the *RepairingCar* aggregated TU. In accordance with the thematic shift in the story (going from shopping to repairing a car), there is a change among the most activated elements of the memory. After clamping the concepts of the first sentence, the most activated elements are part of the *GoingSupermarket* TU. On the contrary, the graphs of the *RepairingCar* aggregated TU are not activated because nothing is mentioned about the situation of repairing. When the second sentence is introduced (after cycle 25), the graphs of the *GoingSupermarket* TU are still highly activated. This is the result of two influence streams. First, the previous activation state of the memory created a context which constrains its evolution. Second, concepts coming from the sentence (in this case, the concept Buy) confirm the previous topic. But we can see also that others concepts (SparePart, Car, Repair) make another topic appear. This explains why the graphs of the *RepairingCar* TU, and the TU itself, begin to be activated. After the processing of the third sentence (after cycle 56), the context effect of the memory for the first topic is not significant any more (the previous confirmation was not very strong) and no element of this third sentence is specifically linked to it. On the contrary, it strengthened the second topic. This explains why we can observe that the activation of the elements of the *GoingSupermarket* TU decreases rather strongly and that the opposite effect happens for the elements of the *RepairingCar* TU.

The spreading activation mechanism which supports this process is divided in two steps. Firstly, it defines a subset of the memory, in which the selection will be done. The episodic memory may be very large if the system has a great amount of experiences. Hence it is not realistic from a computational point of view to involve the whole memory in the core selection mechanism. Moreover, this could be a source of noise and disturb the selection process. It is of course necessary in this task to bypass some similarity failures: the above little story does not tell us about a supermarket, yet the *GoingSupermarket* TU, which is the best thematically related TU, has been nevertheless retrieved. This is due to the spreading of the activation flow towards more general or more specific concepts than the ones initially clamped. But this flow has to stay around the initial concepts. Thus it is not interesting to select a surtype of Supermarket as PublicPlace because we do not want to activate all the situations which take place in a public place.

The definition of this subset of the memory is based on the propagation of a constant flow of activation in the network, a flow that starts from the concepts that make up the clauses of the texts. As the global amount of activation that can be used

for the propagation is always the same, the activation level of the units is less and less high as these units are more and more distant. When the activation level of an unit is under a given threshold, the propagation towards its next units is stopped. In the lattice of concept types, a specific mechanism makes more activation going towards subtypes of a concept type than towards its surtypes.

The second step of the spreading activation mechanism aims at selecting the aggregated TUs in the episodic memory that are the most relevant with regard to the clauses which are considered at a given stage of the analysis of a text. This process is akin to the *evidential activation* in REMIND. When this working mode is active, the network mentioned above, which is a recurrent one due to the symmetry of the connections, has a two-phases dynamics: at the beginning, it has great liberty to explore the space of possible states and this liberty is restrained progressively in order to make the network converge on one state. This is achieved locally by the activation function of the units through its divisor term:

$$A_i(t+1) = \frac{\sum_j w_{ij} \cdot A_j(t)}{t^2} + A_i(t)$$

with $A_i(t)$: activation level of unit i after t cycles
w_{ij}: weight of the connection between units i and j

The second term of this function is particularly significant since it is the support of the context effect mentioned above which makes the network take into account the result of the previous spreading activation sessions.

5 Discussion

5.1 Processing of texts

We give here a complete example processed by MLK. The memory contains in particular the TU *Attempted_Murder* described in figure 3 resulting from the processing of five texts. The creation of the TU comes from the first episode, Airport_Murder_Attempt because none similarity has been found with existing TUs. The other texts, a Quarrel [2], two Politic_Murders [3] and [4] and an Aggression [5], contain a situation matching with the evolutionary aggregated *Attempted_Murder* TU. With the episodes [2] and [3], the similarity results from a deep comparison of the graphs while within the two last episodes, it results more from the global measure. Thus, the more the TU is growing, the more the similarity becomes obvious. The new processed text is given in figure 1. We will focus on the processing of the TU MLK_Attempted_Murder. Each graph causes a propagation and entails highly activating the *Attempted_Murder* unit. The Circumstances and Outcomes slots are strongly similar while the Description slot has been found similar after a deep evaluation. Therefore aggregation is done; results are given in figure 5.

The aggregated TU has been enriched by new events. *Located* has been reinforced while *Mad* is added in the Circumstances slot. In the Description slot, *Stab* is reinforced and *Take_Hospital* is added. In the third slot, *Wounded* is reinforced and *Patient* added. A new causal relation appears

Circumstances	
(a) Located **(0.33)** [1,6]	***(b) Disagree*** **(0.33)** [2,3]
(c) Live **(0.33)** [3,5]	***(d) Have_Bath*** **(0.16)** [3]
(e) Threatened **(0.16)** [3]	***(f) Commanding*** **(0.16)** [4]
(g) Sleeping **(0.16)** [5]	***(h) Mad*** **(0.16)** [6]
Description	
(a) Attack **(0.33)** [1,5]	***(b) Stumble*** **(0.16)** [1]
(c) Stab **(1.0)** [1,2,3,4,5,6]	***(d) Arrest*** **(0.33)** [1,2]
(e) Hit **(0.16)** [2]	***(f) Enter*** **(0.33)** [3,5]
(g) Lose **(0.16)** [4]	***(h) Attach*** **(0.16)** [5]
(i) Tear **(0.16)** [5]	***(j) Take_Hospital*** **(0.16)** [6]
Outcomes	
(a) Imprisoned **(0.5)** [1,2,5]	***(b) Wounded*** **(0.5)** [1,5,6]
(c) Dead **(0.5)** [2,3,4]	***(d) Guillotined*** **(0.16)** [3]
(e) Patient **(0.16)** [6]	

Figure 5: A resulting TU after the aggregation process

between *Take_Hospital* and *Patient*. Comparison of the weights before and after aggregation, shows that they increase with reinforcement while decreasing in the opposite case. Even with only six examples, a structure begins to emerge and the overall frame becomes meaningful. Our method groups events that are thematically linked and, even if misunderstandings occur when segmenting texts, the aggregation will fix them in the long term. Aggregation provides a strong process where some noises are allowed.

5.2 Towards abstraction

At this state of structuration, we can notice in the evolution of the aggregated TU some indications about further abstraction. Strongly weighed events will belong to a general situation, as *Stab*; events as *Arrest* have to be confirmed. Events such as *Dead* and *Wounded* are not simultaneously present in the different episodes but are causally linked to the same action. This will suggest creating a generalized event for them in the general structure, with two specializations including these specific events. Thus, an aggregated structure will lead to create schemata hierarchically structured.

Generalization of concepts also prepares the abstraction phase. Information that will be used are weights, but also the distribution of the episodes and the aggregated relations.

5.3 A reasoning medium

The episodic memory is build for the purpose of learning and improving the comprehension. In this latter perspective, we have developped a thematic analysis that segments a text into TUs by using the memory and its selection mechanism [Ferret and Grau, 1997]. The accumulated experience of MLK is then reused to process new texts.

Enrichment of TUs will allow to improve understanding and make inferences. For example, in the fifth episode, the fact that the aggressor is imprisoned is explicitly mentioned. Because this kind of fact has been memorized as the outcome of the *Arrest* event with an important weight, the arrest of the aggressor could be inferred. Such an inference can be justified because the roles are identical in the specific situations. MLK makes this kind of reasoning possible due to the case basis dimension of its memory.

6 Comparison with other models

AQUA is the closest system to MLK in terms of their goals. It learns explanation patterns (XPs), even if they are incomplete. These XPs are completed when the lacking explanations are found in further texts. But the two processes are quite different. AQUA needs a strong domain theory to build its explanations. This knowledge is represented by MOPs and abstract XPs. AQUA learns new specializations of these XPs. Learning and generalization occur at each new case. Generalization relies on the explicative structure even if some actions need further explanations. In MLK, to avoid the need of general descriptions in all domains, causal links come from linguistic clues, from the assumption that texts are coherent by themselves and from semantic knowledge. If these links are recurrent, they will be confirmed. On the other hand, if they are unusual or result from a misinterpretation, they will disappear by non-reinforcement. So generalization is done in two steps. The aggregation principle allows relevant events to appear with their inter-relations. It replaces the explanation mechanism needed to justify events and solves the incremental aspect of learning. The second step of abstraction, where events and roles are abstracted to produce general situations, must be realized by another process, when situations in the episodic memory are stabilized.

Case retrieval is quite different in MLK and problems of misindexation solved by AQUA do not occur. It is closer to REMIND which integrates episodic memory retrieval and comprehension in the same propagation process. Comparison is only done on the episode retrieval results, since comprehension is quite different when dealing with incomplete knowledge. Differences come from the integration and the structuration of episodes. First, updating of the weights in MLK is done automatically. Second, frames in REMIND are intermediate between our semantic graphs and the TUs. Episodes in REMIND are analoguous to our TUs but TUs are much more structured units where all the frames explaining an episode are grouped. In MLK, weights on events code the importance of the frame in the episode. Thus, this upper level affects the spreading of the activation and selects the TUs not only superficially (as they can be in REMIND) but also thematically close.

7 Conclusion

In this article, we have presented MLK, an episodic memory model which has been designed to support a comprehension process tightly tied to learning. In this model, text representations built by the comprehension process are stored in memory in order to be used later by this same process to analyze other texts. But unlike traditional CBR systems, the memorization in MLK does not consist in classifying a new element according to an existing frame. By aggregating the text representations, the similar situations to which they refer form aggregated units containing events having a weight which characterizes their recurrence degree. This allows a new sort of knowledge to emerge. Weights are used in order to support the retrieval process in the memory: a spreading activation mechanism selects the aggregated units that are the most relevant according to both the current input and the context set by the previous inputs. So, even when a strong domain theory is absent, MLK is able to memorize and organize the result of a comprehension process in such a way that it is able to recall it later in a contextually relevant way. Moreover, this knowledge evolves progressively as new text representations are memorized.

At its present state, our memory model is implemented in Smalltalk and has been tested with a set of 1939 conceptual frames (concept types and the graphs associated to them) and 19 aggregated TUs resulting from 26 texts, i.e. 65 TUs. A linguistic based module whose goal is to build the internal representation of the TUs and a generalization process are being studied. So, in the future, we intend to integrate all these aspects in order to build a complete system integrating learning and comprehension.

References

[Bartlett, 1932] Frederic C. Bartlett. *Remembering.* Cambridge University Press, Cambridge, 1932.

[Ferret and Grau, 1995] Olivier Ferret and Brigitte Grau. An episodic memory for understanding and learning. In *Proceedings of RANLP'95*, pp. 221-229, Bulgarie, 1995.

[Ferret and Grau, 1997] Olivier Ferret and Brigitte Grau. Une analyse thématique s'appuyant sur une mémoire épisodique. In *Proceedings of JST Francil 97*, pp. 161-168, France, 1997.

[Kolodner and Simpson, 1989] Janet Kolodner and R. L. Simpson. The MEDIATOR: Analysis of an early case-based problem solver. *Cognitive Science*, 13(4):507-549, 1989.

[Lange and Wharton, 1993] Trent E. Lange and Charles M. Wharton. Dynamic Memories: Analysis of an Integrated Comprehension and Episodic Memory Retrieval Model. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence,* pp. 208-213, France, 1993.

[Pazzani, 1988] Michael J. Pazzani. Integrating explanation-based and empirical learning methods in OCCAM. In *Proceedings of the 3rd European Working Session on Learning*, pp. 147-165, Glasgow, 1988.

[Ram, 1993] Ashwin Ram. Indexing, elaboration and refinement: incremental learning of explanatory cases. *Machine Learning*, 10(3):7-54, 1993.

[Schank, 1982] Roger C. Schank. *Dynamic Memory: a Theory of Reminding and Learning in Computers and People*. N.Y. Cambridge University Press, 1982.

[Schank and Leake, 1989] Roger C. Schank and David B. Leake. Creativity and Learning in a Case-Based Explainer. *Artificial Intelligence*, 40(1-3):353-385, 1989.

[Sowa, 1984] John F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison Wesley, 1984.

[Vygotsky, 1962] L.S. Vygotsky. *Thought and Language.* MIT Press, Cambridge-Mass, 1962.