Chapter 11
Computer-Supported Human Creativity and Human-Supported Computer Creativity in Language

Lorenzo Gatti, Gözde Özbal, Marco Guerini, Oliviero Stock, and Carlo Strapparava

Abstract This chapter is concerned with human–computer collaboration to achieve linguistic creativity. We claim that humans and computers may benefit from each other during the creativity process and we demonstrate concrete examples of systems that allow different degrees of interaction with the user. Then, we focus on computer-supported human creativity, where the computer necessarily requires human intervention, either for providing input or decisions that are essential to the system, or for deciding which outputs are interesting since the computer lacks a quality metric. As examples of systems modeling computer-supported human creativity, we describe GRAPHLAUGH, an interactive system which produces humorous puns by modifying familiar expressions, and SUBVERTISER, a mobile application that allows users to creatively alter the message contained in pictures of posters, billboards, and advertisements. Finally, we focus on human-supported computer creativity, where the burden of the creative process is mainly on the computer, while the human simply mediates the process during key steps whenever required. As an example modeling this type of creativity, we introduce HEADY-LINES, which automatically generates creative headlines combining a well-known expression with a concept from the news.

Lorenzo Gatti
FBK-irst, Trento, Italy.
Current address: University of Twente, The Netherlands. e-mail: l.gatti@utwente.nl

Gözde Özbal
FBK-irst, Trento, Italy. e-mail: gozbalde@gmail.com

Marco Guerini
FBK-irst, Trento, Italy. e-mail: marco.guerini@fbk.eu

Oliviero Stock
FBK-irst, Trento, Italy. e-mail: stock@fbk.eu

Carlo Strapparava
FBK-irst, Trento, Italy. e-mail: strappa@fbk.eu

T. Veale, F. A. Cardoso (eds.), *Computational Creativity*, Computational Synthesis and Creative Systems, https://doi.org/10.1007/978-3-319-43610-4_11

11.1 Introduction

Although creativity is a fundamental attribute of the human species, the increasing capabilities of computer hardware and software are starting to challenge the assumption that creativity cannot exist outside the boundaries of human minds. Currently, human–computer collaboration is commonly used to achieve creativity: many creative systems distill the useful information from available resources that might be too broad for an unaided human mind, while hiding all the inner complexity of the process. Users are typically involved in the creative process, both to compensate for the increased complexity of higher-level creativity and to fully exploit inherent qualities of both humans and computers (Özbal, 2013).

Let us briefly explore human–computer collaboration with reference to the classification of types of creativity provided by Margaret Boden (2009).

Human creativity may benefit from the support of computer systems, especially in the case of what Boden called combinatorial creativity. This type of creativity involves the combination of familiar notions, concepts, and ideas in an unfamiliar way. The juxtaposition of seemingly unrelated concepts, as in an oxymoron, is an example of combinatorial creativity. Creative humans may be helped by a system that assists them, providing a view of the relevant search space at various moments of the creative process. Ideally a system should really understand what these phases are, understand the needs at each moment, and offer an interface that allows humans to have support without being distracted from their mental activity, but, on the contrary, be able to focus better without interference. In theory, one could also think of intelligent suggestions coming from the exploration of fields different from the one under examination.

Exploratory creativity is instead based on the exploration of some well-defined space, where new concepts and ideas can be combined through a generative process. The exploration of the resulting space can also lead both to a better understanding of the limits to which the generative process can be pushed, and to the discovery of regions that have not been considered already.

The third type of creativity in Boden's classification, transformational creativity, is based on alteration of the rules and the geometry of the conceptual space, which can open up the potential for new forms of expression that would not be possible within the rule set of the original space. It does not seem realistic to suggest this type as an opportunity for a system that supports human creativity, perhaps with the exception of offering excellent editing facilities, where human expressivity can easily pass through a cycle of hypothetical creative solutions and fast adjustment of novel ideas.

Another theory that has proved very influential for the field of computational creativity was developed by Fauconnier and Turner (2008). The conceptual blending theory describes one of the basic mechanisms of the creative process, where novel creations are obtained by merging elements and relations that normally belong to different scenarios. The reader can find concrete examples and a focused overview of computational approaches to conceptual blending in other chapters appearing in this book (Martins, Pereira, & Cardoso, 2019; Veale, 2019).

Not only can systems support humans during the creativity process, but it is also possible for a system to utilize the assistance of users, especially for the production of the first two types of creativity (transformational creativity seems to be beyond the limits of automation, at least for the time being). In this case the system conducts the creative process itself, all the way to the output of a specific result. The role of the human is possibly (i) to restrict the search space of the system; (ii) to select those partial solutions that appear most promising or relevant; and (iii) to simply validate the best of the final results, if they cannot be ranked autonomously or if a threshold of quality for human appreciation is impossible to define and to measure against the solution at hand.

The importance of human-supported computer creativity can be well appreciated if one thinks of adaptive creativity, where the goal is to produce different solutions for different sets of people, individuals, or situations. When a vast amount of content needs to be adapted, or when the input (or target of the communication) is changing continuously, it is inconceivable that a human, even if supported in their activity by a system, can produce such solutions. However, a computer can do the job instead, possibly being guided in its search space at key points or receiving feedback on some of the solutions offered.

For linguistic creativity, adaptivity can be very important: consider personalized advertisements, user-adapted humor, irony, and poetry, to mention just a few themes. The potential is simply enormous considering the huge amount of textual content we are exposed to and its constant transformation.

In this chapter, we describe our experience in building human-supported systems that display various forms of linguistic creativity.

We first summarize our early explorations of computational humor, in particular GraphLaugh, a system for the generation of homophonic puns. Then, we introduce Subvertiser, a mobile application that allows users to spoof pictures of billboards, replacing their textual content with content automatically modified by a creative system to produce an ironic effect. Finally, we describe Heady-Lines, a system that takes existing well-known expressions and innovates with them by bringing in a novel concept coming from evolving news. The resulting sentences can be used as creative headlines or adaptive slogans.

In particular, the focus will be on "human-supported computer creativity," i.e., where there is a limited role for human intervention and the burden of the creative process is mainly on the computer, with humans simply mediating the process at key steps.

Computer-supported human creativity focuses on human creativity and computer systems that can facilitate some part of the creative process, for example by providing humans with suggestions.

Human-supported computer creativity is, instead, centered on the creativity of computers, with humans assisting the computer only when necessary, for example by correcting mistakes or by expanding the search space.

11.2 Related Work

Research in creative language generation has thrived in recent years and state-of-the-art computational models of creativity often produce remarkable results, for example those developed by Manurung et al. (2008), Greene, Bodrumlu, and Knight (2010), Guerini, Strapparava, and Stock (2011), Colton, Goodwin, and Veale (2012), to name just a few. Here we introduce some of the systems that have appeared in the literature, focusing in particular on the distinction between noninteractive and interactive systems. For a review of design principles for creativity support tools in well-established areas of industrial interest, the reader may refer to Shneiderman (2007).

11.2.1 Noninteractive Systems

On one side, we have creative systems that require no manual intervention. For example, Lessard and Levison (1992) generate puns consisting of a quoted utterance and an adverb, by finding a configuration of a root word which can be in an adverb form and a sentence semantically linked to this root word (e.g., "'Turn up the heat,' said Tom coldly").

WISCRAIC (Witty Idiomatic Sentence Creation Revealing Ambiguity In Context) creates jokes with the focus on witticisms (i.e., clever and often ironic remarks) based on phonological ambiguity by extracting semantic associations from both the normal context of words and humor-independent lexical entries (McKay, 2002).

The METAPHORISMYBUSINESS[1] "Twitterbot" is a metaphor-generating program that uses the web service described by Veale (2014) to create novel metaphors and publishes them on the Twitter social platform, one every hour.

With all these systems, the user is only "required" to judge, from among the many possible outputs, which ones are particularly interesting.

11.2.2 Minimally Interactive Systems

Other prototypes require a limited amount of user input, either some material to use as a basis for the creative process, or some parameters with which to modify it. This input is typically required during the first steps of the algorithms.

For example, Stock and Strapparava (2003) use semantic field opposition, rhyme, rhythm, and semantic relations to parody an existing acronym, or to generate a new one for a concept provided by the user.

---

[1] The bot's output is published at https://twitter.com/MetaphorMagnet, and a short description can be found at http://prosecco-network.eu/event/metaphormagnet-creative-metaphor-generating-twitterbot.

STANDUP (Manurung et al., 2008) is a riddle generator that attempts to create a language playground for children with complex communication needs. Children can choose a word from a list, and the system then uses it as the basis for a riddle generated in real time. Despite the positive impact on the children using this system, the overall quality of the jokes has not been evaluated.

Greene, Bodrumlu and Knight (2010) describe a model for poetry generation in which users can control the meter and rhyme scheme. Generation is modeled as a cascade of weighted finite state transducers that only accept strings conforming to the desired rhyming scheme.

Colton, Goodwin and Veale (2012) present a data-driven approach to poetry generation, based on simile transformation. While the user can change some constraints that determine how words are selected (such as their phonetic properties or frequencies), these authors built a system where they "handed over the high-level control," letting daily news influence the mood and theme of the poems. The system also integrates esthetic appreciation mechanisms and provides a commentary on its own work.

In the system developed by Toivanen et al. (2012), the user can specify a topic that is used to generate novel poems by replacing words in existing poetry with morphologically compatible words that are semantically related to the topic.

Possibly closer to a slogan generation system, VALENTINO (Gatti, Guerini, Stock, & Strapparava, 2014) can modify existing textual expressions to obtain more positively or negatively valenced versions. The user provides a sentence and a target score, indicating whether the sentence should become more positive or more negative, and the system then adds, replaces, or deletes words of the original text taking into account their valence, grammatical, and syntactical constraints.

Finally, BRAINSUP (Özbal, Pighin, & Strapparava, 2013), an extensible framework for the generation of creative sentences for educational and advertising applications, lets users force several words to appear in those sentences. BRAINSUP makes heavy use of syntactic information to enforce well-formed sentences and to constraint the search for a solution, and provides an extensible framework into which various forms of linguistic creativity can easily be incorporated.

11.2.3 Interactive Systems

The last type of creative system is the one where the user can intervene during many different steps of the creation process. The computer, in this case, acts as a digital colleague for its human partner (Lubart, 2005), and collaboration between the two is the key to successful creative work.

While this is common in many creative systems outside the linguistic domain, for example in music (Bell & Gabora, 2016), graphics (Davis et al., 2015), and even dance choreography generation (Carlson et al., 2016), it seems that most programs for the generation of text usually allow a lesser degree of interactivity. Still, some examples of truly interactive systems can be found in this domain too.

One such prototype is NAMELETTE (Özbal & Strapparava, 2013), a system for generating creative names. The user can choose a category of products and a set of properties to be underlined. The system finds related concepts and qualities associated with the product, and uses them to generate a name (either a homophonic pun, a metaphor, or a neologism produced by adding a Latin suffix to an English word). During this process, both semantic appropriateness and sound pleasantness of the generated names are taken into account. The user can also intervene in the middle of the generation step, by filtering the concepts and qualities retrieved by the system or adding new ones. As an example of the output of the system, the three homophonic puns generated for an Italian restaurant are *eatalian* (from the combination of "eat" and "Italian"), *pastarant* ("pasta" + "restaurant") and *peatza* ("pizza" + "eat"). As another example, for a cool and sporty brand of sunglasses, NAMELETTE suggests the Latinized names *darkissima*, *polarizium*, and *eyelogia*.

GRAPHLAUGH (Valitutti, Strapparava, & Stock, 2009) (discussed in greater detail in Section 11.3.1) presents an interactive system which generates humorous puns obtained through variation of familiar expressions. The system shows a dynamic graph where the user can choose to start from different familiar expressions, select the replacement words, and tweak other aspects of the generation process while it is still happening.

It is worth noting that, even though fully interactive systems let the user intervene during different stages of the process, they do not necessarily force the user to do so. HEADY-LINES (Gatti, Özbal, Guerini, Stock, & Strapparava, 2015; Gatti, Özbal, Guerini, Stock, & Strapparava, 2016), a system for generating creative headlines for an article, can work in an interactive way, asking the user to select a news article and intervene during the process and select the best headline, but is also capable of working in a fully automatic way, providing a small number of generated headlines for articles in a database and ranking these according to its own output-quality metric. This last prototype, and the distinction between interactive systems that can work both with and without the user, will be the focus of Section 11.4.

11.3 Computer-Supported Human Creativity

In this section, we present a few systems that tackle the problem of supporting humans in the production of creative linguistic content. Such systems are built for different tasks (such as the generation of humorous language or poems), and they support different degrees of interactivity with the user. What characterizes computer-supported creativity is the fact the systems concerned are based on computer tools that may facilitate some phases of the creative production process. They may include subsystems that suggest appropriate words or expressions, systems that may help focus on something and so restrict the creative search space, or, in principle, even tools that automatically assess the value of a certain creative production. Tools of this kind, depending on their cleverness and the quality of the computer–human interface, may be very helpful and ease the production process. Human creativity remains the main contributor to the result, and that tends to provide a final human-level quality. From an applied point of view, the support of such tools is valuable in cases where one single final output expression is sought. If, instead, the application scenario requires many different productions, possibly in parallel, then it is less attractive, because human creative intervention tends to have a bigger role in the final phase of the production, when different linguistic expressions need to be realized. As we shall see in the next section, this is potentially the case when we take into account adaptivity of the outcome to different audiences if they must be reached at the same time or if we have time constraints, for instance in the case where we want to produce messages that change, continuously, on a short time scale.

11.3.1 GRAPHLAUGH

GRAPHLAUGH was one of the earliest attempts to build an interactive system for producing humorous puns through variation (i.e., word substitution) of familiar expressions (Valitutti et al., 2009). The replacement word is selected according to phonetic similarity and semantic constraints, expressing semantic opposition or evoking ridiculous traits of people.

The system can generate puns such as “Chaste makes waste” (a variation on the proverb “haste makes waste”) and “Genital Hospital” (a variation on “General Hospital,” a soap opera title).

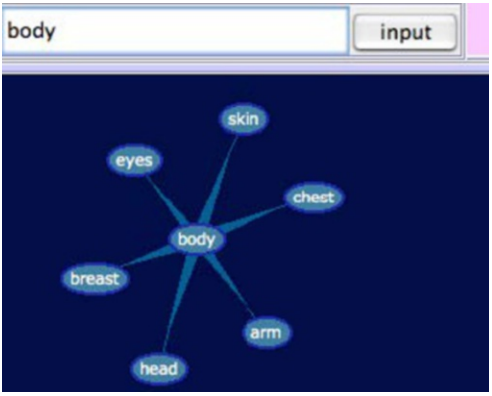
GRAPHLAUGH can automatically generate different types of lexical associations and visualize them in a dynamic graph. Through interaction with nodes and arcs of the network, the user can control the selection of words, semantic associations, and familiar expressions to direct the creative process.

To create its puns, the system requires:

- A list of familiar expressions. A set of 1836 expressions, recognized as “familiar” by English speakers, was collected from the web. It consists of 628 proverbs in common use, 290 famous movie titles, and 918 clichés.
- Phonetic distance. The information on the mapping between words and their phonetic transcription relies on the CMU Pronouncing Dictionary.[2] The algorithm for measuring the phonetic distance is a specific implementation of the Levenshtein distance (Levenshtein, 1966). It is based on a sequence of elementary operations applied to the phonetic expression for a word in order to obtain another word. The weight associated with the substitution operator was modified to take into account the phonetic type, tonic accent, and vowel length.
- Semantic associations. The generation of semantically similar words is based on a measure of lexical similarity. To obtain a vector representation of words, latent semantic analysis (LSA) based on the British National Corpus (BNC)[3] was used. As a measure of semantic similarity between words, GRAPHLAUGH

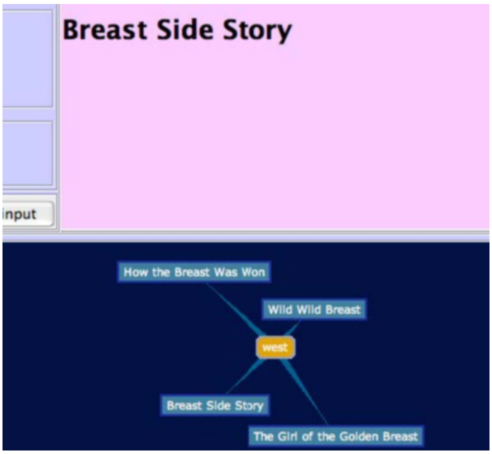
[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[3] http://www.natcorp.ox.ac.uk/



(a)

(b)



(c)

Fig. 11.1 The GRAPHLAUGH interface: (a) selection of a concept to explore; (b) selection of the new word to be used; (c) selection of the preferred pun.

uses the distance in the resulting vector space, measured as the cosine of the angle between the corresponding vectors. The system also takes into account additional semantic constraints, such as antonymy or semantic domain opposition. These are useful for detecting different types of lexical incongruity. In particular, GRAPHLAUGH uses the *Affective-Weight* function (Strapparava, Valitutti, & Stock, 2006) to assign an affective rating to words, so that the system can combine words with different polarity.

- Dynamic graph. GRAPHLAUGH uses the TouchGraph (Alani, 2003) library to create a dynamic graph that stimulates users to explore a network of concepts and expressions. During the interaction, only the currently selected node and a number of adjacent nodes are visualized. This way, the user is free to explore creative local associations without paying attention to the overall structure.

Initially, the user is required to input a concept from which the creative exploration will start (e.g., "body" in Fig. 11.1a). This concept is expanded using LSA similarity and WordNet relations (such as hyperonymy) to produce a list of candidates that could be used in a pun (e.g., "skin", "arm", "breast", ...). After the user has selected one, a graph showing the words from the familiar expression that can be replaced by the candidate is shown (Fig. 11.1b). Finally, the possible puns constructed are shown and the user can select the best one (Fig. 11.1c) .

GRAPHLAUGH clearly is a system for computer-supported creativity, since every step is determined by the user, from the initial concept selection down to the choice of the pun. The system here is just assisting the human creativity, providing some suggestions in a restricted (and usually promising) part of the whole search space, without the ability to run automatically, since the algorithm cannot decide which alternatives are better.

11.3.2 SUBVERTISER

Another example of computer-supported creativity can be seen in SUBVERTISER, a mobile application that allows users to alter messages on posters, billboards, and advertisements by choosing from among creative suggestions proposed by the system. The rationale behind this "virtual defacement" is to help users fight persuasive advertising in an ironic and nondestructive way.

In a typical usage scenario, users are walking with friends in a city, perhaps shopping or going to see a movie. When they notice a billboard that bothers them, they can start our application and use it to produce a new virtual version of the advertisement with the same layout and visual aspect, but with a creative variation of its existing message.

However, this production mechanism is not entirely automatic. The user has to provide the initial material (i.e. the picture that will be modified), and select a message from a small list of suggestions automatically produced by the system. This message, when substituted for the original one, is expected to result in a pleasing effect.

(a) (b) (c)

Fig. 11.2 The main steps of SUBVERTISER: (a) text area selection; (b) modified text selection; (c) final output.

Let us briefly describe the algorithm behind SUBVERTISER. When the program starts, the user is asked either to take a picture with the camera in the phone, or to choose a picture from an image gallery. After that, the user needs to identify the region of text to be modified – in our example, "no tea was ever so health friendly" – by moving and resizing a selection rectangle (Fig. 11.2a).

The Tesseract OCR[4] program is then launched on this area, recognizing the text of the message. The OCR program also detects the coordinates of the bounding boxes of every individual word. SUBVERTISER then takes the rectangle containing the first line of text, downscales it to 100 pixels in height, and uploads it to the WhatTheFont[5] font recognition service, using dedicated APIs.

To detect the color of the text, we cluster the pixels of the first line of text into two classes using K-means clustering. The mean of the smaller class is then taken as the color of the text.

Meanwhile, the program also applies the OpenCV "inpainting" algorithm to each line of the original text, to reconstruct the background image that was underneath it. This gives us a clean background where new text can be superimposed.

[4] http://code.google.com/p/tesseract-ocr/

[5] http://www.myfonts.com/WhatTheFont/

After the user has reviewed the message identified by the OCR system to correct any detection errors, this text is sent to a server running VALENTINO (Gatti et al., 2014), a natural language processing system for automatically changing the affect of a short message. For example, given the word "tea," VALENTINO can describe it as "delicious" or "humble," depending on how positive the final output should be. For SUBVERTISER, we ask this system to produce four different valenced sentences, which are then presented to the user, from the most positive to the most negative (Fig. 11.2b).

Once the user has selected one of the computationally created messages ("no hallucinogenic tea was ever so health friendly" in our example), SUBVERTISER determines how to divide the "slanted" text into lines. Since our text modification system not only replaces sentiment-bearing words but also often inserts or removes them, heuristics are used to ensure that the overall text layout is as similar as possible to the original one. The same goes for capitalization, so if a word was capitalized, its replacement will also be capitalized, and the same happens if a new word is inserted into an all-capital chunk.

Then, we use the WhatTheFont service to generate an image of the new text with the same font and color as the original one, and a transparent background.

Finally, this image, with the message selected by the user, is copied inside the bounding boxes of the original text, as detected by the OCR program. The image is shown on screen (Fig. 11.2c) and the user can save it to the image library or share it. More examples of the output of the system are presented in Table 11.1.

Table 11.1 Advertisements modified with VALENTINO

Original text	No tea was ever so health friendly
Slanted text	No hallucinogenic tea was ever so health friendly
Original text	We thought people would want a different kind of car. One that wasn't so much a car.
Slanted text	We thought rude people would want a different kind of car. One that wasn't so much a nice car.
Original text	The manliest low-calories soda in the history of mankind
Slanted text	The manliest low-calories soda in the history of impotent mankind

To sum up, the process of SUBVERTISER requires that the user assists the system in:

1. Providing of a photograph of an interesting advertisement.
2. Indicating which area of the picture should be analyzed by OCR.
3. Correcting the potential mistakes of the automatically recognized text.
4. Selecting a good automatic variation of the original message, as produced by VALENTINO.

While the third point could reasonably be automated without degrading the quality of the results (e.g., by using a spellchecker, especially if combined with a dedicated language model), meaningful results can only be obtained by considering the interaction between the qualities of the product depicted, the meaning of the text in the advertisement and its modifications, and, possibly, the visual properties of the image. Since SUBVERTISER has no built-in appreciation mechanism for these aspects, it necessarily depends on human intervention and can only work as a “support tool” that suggests potentially useful text modifications.

11.4 Human-Supported Computer Creativity

While creative systems such as those described in the previous section can be very useful for enhancing human creativity, there are scenarios where linguistic creativity is needed (or at least desirable) but the amount of data to be considered in the creative process would make a substantial human contribution infeasible. In these cases, letting the computer guide the process is a necessity.

Two examples of such scenarios come to mind. Personalized advertisements that adapt the message to the user, where the number of recipients would make it impossible to create customized versions manually, could surely benefit from a system that can decide what is the best slogan for any given user. Another case is the creation of catchy headlines, where human intervention can hardly keep up with the flow of freshly produced news. For this particular scenario, a useful tool would be a creative system that can work autonomously – producing multiple headlines for each article and selecting only the best one, so that most articles could get a custom creative title – but that also allows user interaction, so that the most important news items (e.g., those appearing “above the fold” on the front page) can be created “in collaboration” with a copy editor.

This last case could be defined as “human-supported computer creativity”: the computer is in charge of the process, and it can produce meaningful results and rank (and, if necessary, discard) them, while still allowing some human interaction during the different steps of the creation process whenever it is needed. For an interactive system to really “master” the creation process, this ranking capability is essential: while SUBVERTISER can often generate humorous results, this is only possible if the user picks a “good” message. The system has no measure of aptness, level of humor or even incongruity with the original message.

A truly human-supported computational system should ideally provide users with a small *ordered* search space where creative results can easily be found. Potentially, it should be able also to produce meaningful results without human intervention, albeit supporting it to further improve the quality of the results. Thus, human-supported computer creativity can also be viewed as a specific subset of mixed-initiative co-creativity (Yannakakis, Liapis, & Alexopoulos, 2014), i.e., the subset in which computational systems can contribute to a large degree to the solution of a problem without performing simple random generation.

In the following section we describe HEADY-LINES, a prototype that shows these characteristics.

11.4.1 Heady-Lines

Heady-Lines (Gatti et al., 2015) is a system for the automatic generation of creative headlines that combine a well-known expression with a concept from the news. The system is inspired by catchy headlines such as “The dark side of the sun” (for an article about the dangers of tanning booths) or “This little LED of mine” (for one describing new LED lightbulbs) that often appear in newspapers.

The system is composed of four main modules that deal with (i) retrieving the news of the day from the web, (ii) extracting keywords from the news and expanding them with relevant related concepts, (iii) pairing the news with well-known expressions using state-of-the-art similarity metrics, and (iv) generating a new headline by merging the well-known expression with a keyword from the news, satisfying the lexical and morpho-syntactic constraints enforced by the expression.

On top of this algorithm, we have developed a web interface (Gatti et al., 2016) that hides the technical details from the users (ideally copy editors) and collaborates with them in the creative task of generating a good headline. However, the system can still work in a fully automatic mode, where just a piece of news (or a feed of news) is given as input and the best headline is presented, according to the selection criterion of the algorithm.

Initially, users are presented a list of short descriptions (about 25 words) of the news of the day. From a technical point of view, the news items are retrieved from the RSS feed of BBC News and from The New York Times through its API. Each entry is composed of a headline, a short description of the article, and other metadata, but only the description is used by the system. As they are downloaded, news descriptions are tokenized and part-of-speech tagged using Stanford CoreNLP (Manning et al., 2014). An example of a description provided by our interface is “By any measure, it has been a year from hell for the European Union. And if Britons vote to leave the bloc, next year could be worse.” (from The New York Times).

Once the user has selected an interesting news event from the list, its description is presented in a new page with its key concepts highlighted (Fig. 11.3). In particular, stop words and irrelevant words fade to gray, while the defining elements for that news are colored differently, depending on their category. At the moment we are differentiating betwen (i) named entities, (ii) important keywords for which we have some knowledge, and (iii) important but unrecognized keywords. Users are also presented with an additional set of related concepts which are derived from the important words recognized in the sentence. The user can remove any of the identified keywords or related concepts from the list just by clicking on them, or click on a word deemed irrelevant to change its status.

We define the importance of each word as the number of times that word appears in a news corpus (23,415 news documents from the LDC GigaWord corpus (Parker, Graff, Kong, Chen, & Maeda, 2011)), divided by the total number of headlines occurring in the corpus (i.e., the probability of the word). Words under a certain threshold are considered as “key concepts.” The “related keywords” that the users see are simply synonyms and derivationally related forms of these words, obtained from WordNet. The named entities are detected by utilizing CoreNLP. In the example above, the system identifies *hell*, *European Union*, *Britons*, and *bloc* as key terms. It expands this list by retrieving concepts such as the noun *Brit* (a synonym of *Britons*) and the adjective *infernal* (from the noun *hell*).

A list of well-known expressions is then presented to the user, with the expressions that are most related to the news appearing at the top. The user can disable some of the entries by clicking on them (Fig. 11.4) The relatedness is calculated using a skip-gram model (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013) trained on the words of the GigaWord corpus. To compare the news with each expression, we construct a vector representation of the former by summing the vectors of its keywords. Similarly, we build a vector representation of each expression based on its words (after removing the stop words). The expressions that do not reach a certain similarity threshold are discarded. This ensures at least a minimum degree of relatedness between the news and the well-known expression. For the above example, the most similar well-known expression is the national anthem God Save the Queen, followed by the song Son of a Preacher Man. The similarity of the sentences is due to *queen* being related to *Britons*, while *God* and *preacher* are related to *hell*.

The list of the new potential headlines is then shown to the user. They are ranked from best to worst, but the user can click on any of the sentences and see a final page with the headline, along with the starting description, to see how fit it is for the news. The sentences are modified by taking into account the lexical and syntactic constraints imposed by the original expression. This is accomplished by using a database of tuples that stores, for each relation in the dependency treebank of the LDC GigaWord corpus, its occurrences with specific "governors" (heads) and "dependents" (modifiers), similarly to the approach of Özbal et al. (2013). For each lemma *w* in a well-known expression, we determine all the words that are connected to *w* by a dependency relation. Then, we calculate how likely it is that each keyword *k* from the news articles that passed the similarity filter can replace a *w* that is the same part of speech. We can then select the slot containing the word *w* to be replaced, and the best keyword *k* for each news article, by simply maximizing this dependency likelihood. In this case a threshold is enforced also, so that sentences that do not reach a satisfactory level of grammaticality are removed. Finally, the morphology of

Fig. 11.3 Key concepts of the news, and words related to them.

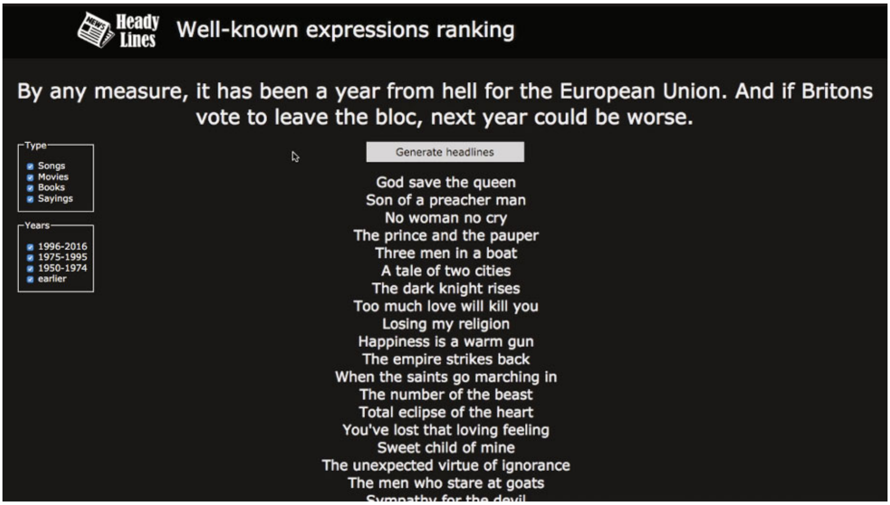


Fig. 11.4 Well-known expressions sorted by similarity to the news.

the replaced word *w* is applied to *k* by using MorphoPro (Pianta, Girardi, & Zanoli, 2008) and the modified sentence is generated. To rank the final output, the system sorts each modified sentence according to its mean rank with respect to similarity and dependency scores, thus balancing the scores for grammaticality and relatedness to the news. The lower the mean, the better the system considers the headline. For our example, the system will offer a very appropriate "God save the bloc" as the best headline. More examples of the output of the system can be seen in Table 11.2.

As we have seen, HEADY-LINES is a system that models human-supported computer creativity. It can automatically produce meaningful and creative content, discarding solutions that are not "convincing" enough according to its own output-quality metric. Nevertheless, the user is allowed to use the system interactively to enhance the output quality.

11.5 Conclusions

The focus of this chapter was the distinction between computer-supported human creativity and human-supported computer creativity. Both are concerned with the production of creative artifacts, but they assign different roles to computers and humans.

In the first case, the computer facilitates part of the creative process, for example by providing humans with suggestions. The system, however, is not entirely autonomous, and human creativity is still key to a successful production.

In the second case, instead, computers can produce creative artifacts on their own, but with the potential for users to intervene during the creative process to enhance

Table 11.2 Output examples

Description	. . . it has been a year from hell for the European Union. And if Britons vote to leave the bloc, next year could be worse.
Expression	God save the Queen
Headline	God save the bloc
Description	Scientists have reconstructed how an ancient reptile swam in the oceans at the time of the dinosaurs.
Expression	The number of the beast
Headline	The ocean of the beast
Description	WTO finally reached deals, capping a ministerial conference . . . where rich and poor countries had been split over the path of trade reforms.
Expression	Bridge over troubled water
Headline	Bridge over troubled division
Description	. . . Australia captain Steve Smith has passed most of the tests presented in leading a team in the throes of transition.
Expression	The empire strikes back
Headline	The captain strikes back
Description	Martin Shkreli resigns as chief executive of Turing Pharmaceuticals following his arrest on Thursday on securities fraud charges.
Expression	Crime and punishment
Headline	Fraud and punishment

the output, for example by correcting the mistakes of the system or by expanding its search space.

We have shown prototypes of systems of both types, and argue that both can be useful in applied scenarios, the former as a creativity-enhancing tool, the latter when a vast amount of content needs to be produced, or if there is a need to adapt a message to a large number of users, or when the inputs are changing continuously, such as when one is producing news or advertisements for the web.

References

Alani, H. (2003). TGVizTab: An ontology visualisation extension for Protégé. In *Proceedings of Knowledge Capture, Workshop on Visualization Information in Knowledge Engineering*.

Bell, S., & Gabora, L. (2016). A music-generating system based on network theory. In *Proceedings of the 7th International Conference on Computational Creativity*.

Boden, M. A. (2009). Computer models of creativity. *AI Magazine*, *30*(3), 23.

Carlson, K., Pasquier, P., Tsang, H. H., Phillips, J., Schiphorst, T., & Calvert, T. (2016). Cochoreo: A generative feature in idanceForms for creating novel keyframe animation for choreography. In *Proceedings of the 7th International Conference on Computational Creativity* (pp. 380–387).

Colton, S., Goodwin, J., & Veale, T. (2012). Full-FACE poetry generation. In *Proceedings of the 3rd International Conference on Computational Creativity* (pp. 95–102).

Davis, N., Hsiao, C.-P., Singh, K. Y., Li, L., Moningi, S., & Magerko, B. (2015). Drawing apprentice: An enactive co-creative agent for artistic collaboration. In *Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition* (pp. 185–186). ACM.

Fauconnier, G., & Turner, M. (2008). *The way we think: Conceptual blending and the mind's hidden complexities*. Basic Books.

Gatti, L., Guerini, M., Stock, O., & Strapparava, C. (2014). Sentiment variations in text for persuasion technology. In *Proceedings of the 9th International Conference on Persuasive Technology* (pp. 106–117).

Gatti, L., Özbal, G., Guerini, M., Stock, O., & Strapparava, C. (2015). Slogans are not forever: Adapting linguistic expressions to the news. In *Proceedings of the 24th international conference on artificial intelligence* (pp. 2452–2458).

Gatti, L., Özbal, G., Guerini, M., Stock, O., & Strapparava, C. (2016). Heady-Lines: A creative generator of newspaper headlines. In *Companion publication of the 2016 International Conference on Intelligent User Interfaces* (pp. 79–83).

Greene, E., Bodrumlu, T., & Knight, K. (2010). Automatic analysis of rhythmic poetry with applications to generation and translation. In *Proceedings of the 15th Conference on Empirical Methods in Natural Language* (pp. 524–533).

Guerini, M., Strapparava, C., & Stock, O. (2011). Slanting existing text with Valentino. In *Proceedings of the 16th International Conference on Intelligent User Interfaces* (pp. 439–440).

Lessard, G., & Levison, M. (1992). Computational modelling of linguistic humour: Tom Swifties. In *Proceedings of the 1992 Joint Annual Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing* (pp. 175–178).

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, *10*(8), 707–710.

Lubart, T. (2005). How can computers be partners in the creative process: Classification and commentary on the special issue. *International Journal of Human–Computer Studies*, *63*(4–5), 365–369.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (pp. 55–60).

Manurung, R., Ritchie, G., Pain, H., Waller, A., O'Mara, D., & Black, R. (2008). The construction of a pun generator for language skills development. *Applied Artificial Intelligence*, *22*(9), 841–869.

Martins, P., Pereira, F. C., & Cardoso, F. A. (2019). The nuts and bolts of conceptual blending: Multi-domain concept creation with Divago. In T. Veale & F. A. Cardoso (Eds.), *Computational creativity: The philosophy and engineering of autonomously creative systems* (pp. 91–118). Springer.

McKay, J. (2002). Generation of idiom-based witticisms to aid second language learning. In *Proceedings of the Twente Workshop on Language Technology 20* (pp. 70–74).

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th Conference on Advances in Neural Information Processing Systems* (pp. 3111–3119).

Özbal, G. (2013). *Computational approaches to linguistic creativity for real world applications* (Doctoral dissertation, University of Trento).

Özbal, G., Pighin, D., & Strapparava, C. (2013). BRAINSUP: Brainstorming support for creative sentence generation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 1446–1455).

Özbal, G., & Strapparava, C. (2013). Namelette: A tasteful supporter for creative naming. In *Companion publication of the 2013 International Conference on Intelligent User Interfaces* (pp. 55–56).

Parker, R., Graff, D., Kong, J., Chen, K., & Maeda, K. (2011). English GigaWord, 5th edn, ldc2011t07. *Linguistic Data Consortium.*

Pianta, E., Girardi, C., & Zanoli, R. (2008). The TextPro tool suite. In *Proceedings of the 6th International Conference on Language Resources and Evaluation* (pp. 2603–2607).

Shneiderman, B. (2007). Creativity support tools: Accelerating discovery and innovation. *Communications of the ACM*, *50*(12), 20–32.

Stock, O., & Strapparava, C. (2003). Getting serious about the development of computational humor. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence* (Vol. 3, pp. 59–64).

Strapparava, C., Valitutti, A., & Stock, O. (2006). The affective weight of lexicon. In *Proceedings of the 5th International Conference on Language Resources and Evaluation* (pp. 423–426).

Toivanen, J. M., Toivonen, H., Valitutti, A., & Gross, O. (2012). Corpus-based generation of content and form in poetry. In *Proceedings of the 3rd International Conference on Computational Creativity* (pp. 175–179).

Valitutti, A., Strapparava, C., & Stock, O. (2009). GraphLaugh: A tool for the interactive generation of humorous puns. In *Proceedings of the 3rd Conference on Affective Computing and Intelligent Interaction* (pp. 634–635).

Veale, T. (2014). A service-oriented architecture for metaphor processing. In *Proceedings of the 2nd workshop on metaphor in NLP* (pp. 52–60).

Veale, T. (2019). From conceptual mash-ups to bad-ass blends: A robust computational model of conceptual blending. In T. Veale & F. A. Cardoso (Eds.), *Computational creativity: The philosophy and engineering of autonomously creative systems* (pp. 71–89). Springer.

Yannakakis, G. N., Liapis, A., & Alexopoulos, C. (2014). Mixed-initiative co-creativity. In *Proceedings of the 9th Conference on the Foundations of Digital Games*.